Knowledge-based scaling for biological models

Anna Zhukova

université de BORDEAUX

THÈSE PRÉSENTÉE

POUR OBTENIR LE GRADE DE

DOCTEUR DE

L'UNIVERSITÉ DE BORDEAUX

ÉCOLE DOCTORALE DE MATHÉMATIQUES ET D'INFORMATIQUE
SPÉCIALITÉ : INFORMATIQUE

Par Anna ZHUKOVA

KNOWLEDGE-BASED GENERALIZATION FOR METABOLIC MODELS

Généralisation de modèles métaboliques par connaissances

Sous la direction de : David James SHERMAN

Soutenue le 18 décembre 2014

Membres du jury :

Mme	JOHNEN, Colette	PR	U. Bordeaux	Présidente
Mme	DEVIGNES, Marie-Dominique	CR	CNRS	Rapportrice
M.	FERTIN, Guillaume	PR	U. Nantes	Rapporteur
M.	DURRENS, Pascal	CR	CNRS	Examinateur
Mme	ROPERS, Delphine	CR	Inria	Examinatrice
M.	SHERMAN, David James	DR	Inria	Directeur de thèse

Titre : Généralisation de modèles métaboliques par connaissances

Résumé : Les réseaux métaboliques à l'échelle génomique décrivent les relations entre milliers de réactions et molécules biochimiques pour améliorer notre compréhension du métabolisme. Ils trouvent des applications dans les domaines chimiques, pharmaceutiques, et dans la biorestauration.

La complexité de modèles métaboliques mets des obstacles á l'inférence des modèles, à la comparaison entre eux, ainsi que leur analyse, curation et amélioration par des experts humains. Parce que l'abondance des détailles dans les réseaux à grande échelle peut cacher des erreurs et des adaptations importantes de l'espèce qui est étudié, c'est important de trouver les correct niveaux d'abstraction qui sont confortables pour les experts humains : on doit mettre en évidence la structure essentiel du modèle ainsi que les divergences de celle-là (par exemple les chemins alternatives et les réactions manquantes), tout en masquant les détails non significatifs.

Pour répondre a cette demande nous avons défini une généralisation des modèles métaboliques, fondée sur les connaissances, qui permet la création des vues abstraites de réseaux métaboliques. Nous avons développé une méthode théorétique qui regroupe les métabolites en classes d'équivalence et factorise les réactions reliant ces classes d'équivalence. Nous avons réalisé cette méthode comme une bibliothèque Python qui peut être téléchargée depuis metamogen.gforge.inria.fr.

Pour valider l'intérêt de notre méthode, nous l'avons appliquée à 1 286 modèles métaboliques que nous avons extraits de la ressource Path2Model. Nous avons montré que notre méthode aide l'expert humain à relever de façon automatique les adaptations spécifiques de certains espèces et à comparer les modèles entre eux.

Après en avoir discuté avec des utilisateurs, nous avons décidé de définir trois niveaux hiérarchiques de représentation de réseaux métaboliques : les compartiments, les modules et les réactions détaillées. Nous avons combiné notre méthode de généralisation et le paradigme des interfaces zoomables pour développer Mimoza, un système de navigation dans les réseaux métaboliques qui crée et visualise ces trois niveaux. Mimoza est accessible en ligne et pour le téléchargement depuis le site mimoza.bordeaux.inria.fr.

Mots clés : modélisation métabolique; généralisation par connaissances; visualisation.

Title : Knowledge-based generalization for metabolic models

Abstract : Genome-scale metabolic models describe the relationships between thousands of reactions and biochemical molecules, and are used to improve our understanding of organism's metabolism. They found applications in pharmaceutical, chemical and bioremediation industries.

The complexity of metabolic models hampers many tasks that are important during the process of model inference, such as model comparison, analysis, curation and refinement by human experts. The abundance of details in large-scale networks can mask errors and important organism-specific adaptations. It is therefore important to find the right levels of abstraction that are comfortable for human experts. These abstract levels should highlight the essential model structure and the divergences from it, such as alternative paths or missing reactions, while hiding inessential details.

To address this issue, we defined a knowledge-based generalization that allows for production of higher-level abstract views of metabolic network models. We developed a theoretical method that groups similar metabolites and reactions based on the network structure and the knowledge extracted from metabolite ontologies, and then compresses the network based on this grouping. We implemented our method as a python library, that is available for download from metamogen.gforge.inria.fr.

To validate our method we applied it to 1 286 metabolic models from the Path2Model project, and showed that it helps to detect organism-, and domain-specific adaptations, as well as to compare models.

Based on discussions with users about their ways of navigation in metabolic networks, we defined a 3-level representation of metabolic networks: the full-model level, the generalized level, the compartment level. We combined our model generalization method with the zooming user interface (ZUI) paradigm and developed Mimoza, a user-centric tool for zoomable navigation and knowledge-based exploration of metabolic networks that produces this 3-level representation. Mimoza is available both as an on-line tool and for download at mimoza.bordeaux.inria.fr.

Keywords : metabolic modeling; knowledge-based generalization; visualization.

Inria / CNRS / University of Bordeaux
joint project-team MAGNOME
[351 cours de la Libération, 33405 Talence Cedex – France]

Table of contents

List of figures

List of tables

List of Algorithms

Chapter 1

Introduction

1.1 Introduction

Fundamental questions in the life sciences can now be addressed at an unprecedented scale through the combination of high-throughput experimental techniques and advanced computational methods from the computer sciences. The field of computational biology or bioinformatics has grown around intense collaboration between biologists and computer scientists working towards understanding living organisms as systems. One of the key challenges in this study of systems biology is understanding how the static information recorded in the genome is interpreted to become dynamic systems of cooperating and competing biomolecules.

Metabolic modeling is a perfect example of these challenges.

1.2 Metabolism and metabolic networks

Metabolism is a mechanism composed by a set of biochemical reactions, by which the cell sustains its growth and energy requirements. It includes several *catabolic* (breaking down large molecules into smaller units) and *anabolic* (constructing molecules from smaller units) pathways of enzyme-catalyzed reactions that import substrates from the environment and transform them into energy and building blocks required to build the cellular components. Metabolic pathways are interconnected through intermediate metabolites, forming complex networks [Palsson, 2006].

Catalysis is the increase in the rate of a chemical reaction due to the participation of an additional substance called a catalyst. *Enzymes* are natural proteins that catalyze chemical reactions.

Metabolic network model is a knowledge construct for modeling of metabolic processes in the cell. It describes molecular species participating in organisms' metabolism and biochemical reactions between them. Metabolic models are used to improve understanding how the *genotype* (set of enzymes encoded by a genome and their regulation) influences *phenotype* (the identity of the molecules that a metabolic network can synthesize, and the rate of synthesis) [Wagner, 2012].

The primary topological properties of a biochemical reaction network are given by *stoichiometry*. The stoichiometry of chemical reactions is fixed and is described by integral numbers counting the molecules that react and that form a consequence of the chemical reaction. Stoichiometry is invariant between organisms for the same reactions and does not change with pressure, temperature or other conditions [Palsson, 2011].

The *kinetic constants*, on the contrary, can vary across a population and change over time through evolution. Even though biological information is growing rapidly, the kinetic information is not always available, especially for genome-scale models.

Metabolic phenotypes can be defined in terms of flux distributions through a metabolic network. *Dynamic analysis* of metabolic flux distributions require kinetic and concentration information about enzymes and various cofactors. For genome-scale metabolic networks, that often lack this information, the *constraint-based* modeling procedure [Bonarius et al., 1997; Edwards et al., 2002] is applicable. It does not strive to find a single solution but rather finds a collection of all allowable solutions to the governing equations that can be defined (a solution space). Solutions that violate any of the imposed constraints are excluded from the solution space. The subsequent application of additional constraints further reduces the solution space and, consequently, reduces the number of allowable solutions that a cell can utilize. The constraints that have been used in the first generation of constraint-based models include stoichiometric constraints, thermodynamic constraints (regarding the reversibility of a reaction), and enzymatic capacity constraints [Reed and Palsson, 2003].

1.3 History of metabolic modeling

The scale of metabolic network reconstructions may range from individual pathways to whole genomes. In 1943 B. Chance published the first numerical simulation of a single enzyme biochemical system, solving the equations for the systems' behavior using a mechanical differential analyzer [Chance, 1943]. Since then metabolic models started emerging.

The advances of genome sequencing led to the advances in modeling. In 1995 the first

complete genomic sequence was obtained, it was a genome of the bacterium *Haemophilus influenzae Rd* [Fleischmann et al., 1995]. This led to the creation of the first genome-scale metabolic model of *H. influenzae* Rd [Edwards and Palsson, 1999] in 1999. It contained 488 reactions operating on 343 metabolites. It was further improved in 2000 [Schilling and Palsson, 2000]. In the following years several other bacteria genome-scale models were created (*Escherichia coli* MG1655 [Edwards and Palsson, 2000], *Helicobacter pylori* 26695 [Schilling et al., 2002]). There did not exist a standard way of model representation back then and these models were mostly encoded as xls files.

In 2003 the need of a standard for model encoding was addressed by the creation of the Systems Biology Markup Language (SBML) [Hucka et al., 2003]. The same year a fist genome-scale model for a yeast *Saccharomyces cerevisiae* [Förster et al., 2003] was created and encoded in SBML. This model contained 1 175 metabolic reactions and 584 metabolites in three compartments: cytosol, mitochondria and extracellular.

As one can see, not only a large increase in the number of computational models in biology was taking place, but also to a dramatic increase in their size and complexity. The number of models deposited in BioModels Database [Li et al., 2010] is doubling roughly every 22 months while the average number of relationships between variables per model is doubling every 13 months [Courtot et al., 2011]. The first release of BioModels Database in 2005 published 30 models. They contained on average 30 relationships per model, and this number rose to around 100 in the 17th release (in 2010) and keeps increasing. The 28th release of BioModels database (September 2014) contains 1 212 models.

1.4 Goals of metabolic modeling

By 2014, all the way from a single-reaction reconstructions to the systematic construction of genome-scale kinetic models [Stanford et al., 2013], containing thousands of reactions, and the first whole-cell computational model [Karr et al., 2012], was done.

Metabolic models found applications in pharmaceutical, chemical and bioremediation industries. The initial applications of metabolic models were in designing *metabolic engineering* strategies that would result in enhanced production of desired target products [Kim et al., 2012]. Current examples include production of food and beverages [Fleet, 2007], pharmaceuticals [Liu et al., 2014], and biofuels [Hollinshead et al., 2014]. See [Copeland et al., 2012] for the review of common tasks encountered by metabolic engineers and the description of relevant computational tools; and [Pitkänen et al., 2014] for an example of comparative metabolic reconstruction of genome-scale network mod-

els for 49 fungal species, including some of the most important production organisms in industrial biotechnology.

The development of genome-scale metabolic models of several pathogenic microorganisms (e.g., *H. pylori* [Thiele et al., 2005], *A. baumannii* [Kim et al., 2010], *B. cenocepacia* [Fang et al., 2011]) lead to their employment for the analysis of diseases and for the discovery of novel *drug targets* suitable for treating the disease. [Chavali et al., 2012] reviews *in silico* strategies to identify effective drug targets, focusing on pathogen metabolic networks. The consensus reconstruction of human metabolism [Thiele et al., 2013] has allowed investigation of human metabolic diseases and simulation of drug actions.

1.5 Metabolic modeling workflow

Metabolic network reconstruction can address various objectives. Examples include creation of a model for a new organism from its genomic data and a reference model for a similar organism; creation of a larger-scale model by combining several models of different aspects of organism's metabolism; improving an existing model by incorporating new data and new expertise. To accomplish these objectives the following tasks are used (see Figure 1.1).

Inference

Metabolic networks for more and more organisms are being inferred and stored in *biological network collections*, such as the Biomodels database [Li et al., 2010], BIGGs [Schellenberger et al., 2010], JWS online [Snoep and Olivier, 2003]. The metabolic network reconstruction process is becoming more and more advanced, and there now exist various tools for semi-automatic model inference, e.g., PathwayTools [Karp et al., 2002], SuBliMinaL [Swainston et al., 2011], CoReCo [Pitkänen et al., 2014]. We describe inference tools in more detail in Chapter 2. During the process of network reconstruction, they infer metabolic reactions from *pathway and reaction databases* such as KEGG [Kanehisa et al., 2012] and Reactome [Milacic et al., 2012], and from existing networks for similar organisms using genomic data [Thiele and Palsson, 2010]. Although automatic model inference tools and genomic comparison methods are becoming steadily more sophisticated, they may still leave gaps in the model or add erroneous reactions. The intrinsic and extrinsic correctness of the model should be checked during the phases of analysis and curation.

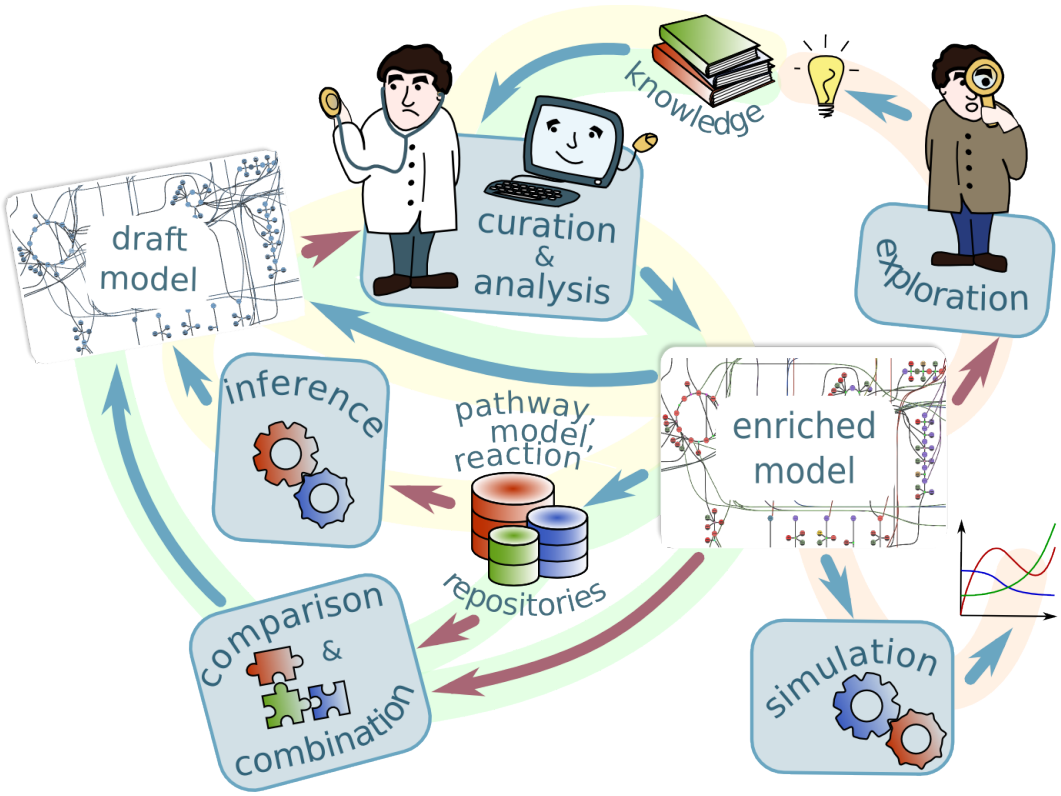

Fig. 1.1 **Metabolic modeling workflow.** The figure shows the processes of metabolic model creation, improvement and usage.
The processes highlighted in yellow represent the *model creation cycle*: The draft model is created by model inference tools based on models for similar organism, pathway and reaction information extracted from model repositories and pathway and reaction databases; it is then iteratively improved during the process of curation and analysis. The resulting model can in its turn be added to model repositories.
The processes highlighted in red show *model usages*: simulation and knowledge-oriented exploration.
The processes highlighted in green describe *comparison and combination of several models*. As the model creation cycle, they also include the curation and analysis stage.
The processes represented with the red arrows can use model generalization, described in this thesis, to discover similarities between the reactions and metabolites in the model, or in different models, and to aid a human understanding of large networks.

Curation and analysis

The inferred draft network needs to be refined during several iterations of analysis, curation and improvement [Swainston et al., 2011; Thiele and Palsson, 2010]. The goal of the *model analysis* is to verify that the model does not contain inner contradictions and errors, e.g., that the network is connected; the transport reactions between compartments

are well defined; the reactions are chemically balanced, etc. Various model analysis tools, e.g., FASTGAPFILL [Thiele et al., 2014] for gap filling, CellNetAnalyser [Klamt et al., 2007] for for finding dead ends and blocked reactions, SuBliMinaL Toolbox [Swainston et al., 2011] for reaction balancing, can facilitate model analysis; but human expert's knowledge on organism's metabolism still plays an important role.

Curation is performed to ensure, first, that all of the knowledge that the experts deem pertinent is recorded in the model, and second, that the knowledge is recorded in a coherent way. The first depends on the requirements of the experts: a model for a cell factory used in an industrial process would need precise kinetics but may only require the reactions active in steady state that participate in the pathway that produces or consumes the target molecule, whereas a whole-genome model used to understand functional dependencies between genes would need to be as complete as possible but may not require reaction kinetics. The second concerns the internal consistency of what is recorded: metabolites and reactions must be annotated with ontology terms from appropriate knowledge bases, reaction stoichiometry must be consistent, transport between compartments must be assured, and so on. Curation and analysis of models is an iterative process, ideally repeated many times to refine the draft model until the needed level of quality is achieved.

The curation by a human expert requires a means of splitting genome-scale models into smaller units that can be checked and analyzed independently. At a higher level, appropriate levels of abstraction need to be found to allow experts to compare whole genome networks. Good model visualization tools are also required.

Simulation

The improved model, created during the iterations of curation and analysis, can be used for computer simulation to obtain numerical results. We do not exploit simulation in this thesis.

Exploration

The model can also be used for knowledge-oriented exploration to obtain new knowledge about the processes happening in the organisms' metabolism, and the relationships between them, e.g., the "redundancy" of the model: discovery of similar reactions, and alternative pathways.

Means of splitting genome-scale models into smaller units, appropriate levels of abstraction and good model visualization tools are as important for model exploration task

as they are for curation.

Comparison and combination

Model comparison and combination is another important task. Possible scenarios include comparison to a different model of the same organism, with potential merging into a new, more complete, model; comparison of a model of a healthy organism to the one of a metabolism suffering from a disease to discover disease-specific metabolic adaptations. A genome-scale model can be created by combining several smaller models, describing different metabolic processes in a species [Schulz et al., 2006], where model comparison is needed to detect overlaps. Such a model can be used as a draft model, and will need to undergo the analysis and curation phase. Finally, a group of models for related species can be compared and combined to produce a concise representation of their common metabolism, to study the common properties of a group, as well as the organism-specific adaptations.

There exist various software facilitating model merging, e.g., semanticSBML [Krause et al., 2010], OREMPdb [Umeton et al., 2012], PathCase-SB Model Composition Tool [Coskun et al., 2013], but all of them require human expert's intervention in cases when the models to be merged are incompatible or contradict to each other, as well as for better discovery of common parts. Thereby, after the creation, the combined model becomes a draft and should in its turn undergo the analysis and curation cycle. We describe model merging tools in more detail in Chapter 2.

By combining these modeling tasks into workflows, as in Figure 1.1, one can accomplish the modeling objectives listed above.

1.6 Understanding genome-scale models

Curation and analysis, exploration, comparison and combination of metabolic models are tasks that involve human experts' work. Human experts, who generally speaking understand best small-sized networks, containing up to hundreds of nodes [Herman et al., 2000; von Landesberger et al., 2011], are distracted by the abundance of details in genome-scale networks (needed for accurate computer simulation) and cannot easily identify the reactions that require their intervention. For example, Figure 1.2 shows the peroxisome compartment of the model of the yeast *Yarrowia lipolytica* (*MODEL1111190000* [Loira et al., 2012]). Even though it contains only 67 reactions (out of 2 002 in the whole *Y. lipolytica* model), it is already quite complicated for a human. A means of splitting

genome-scale models into smaller units that can be checked and analyzed independently by human experts is required.

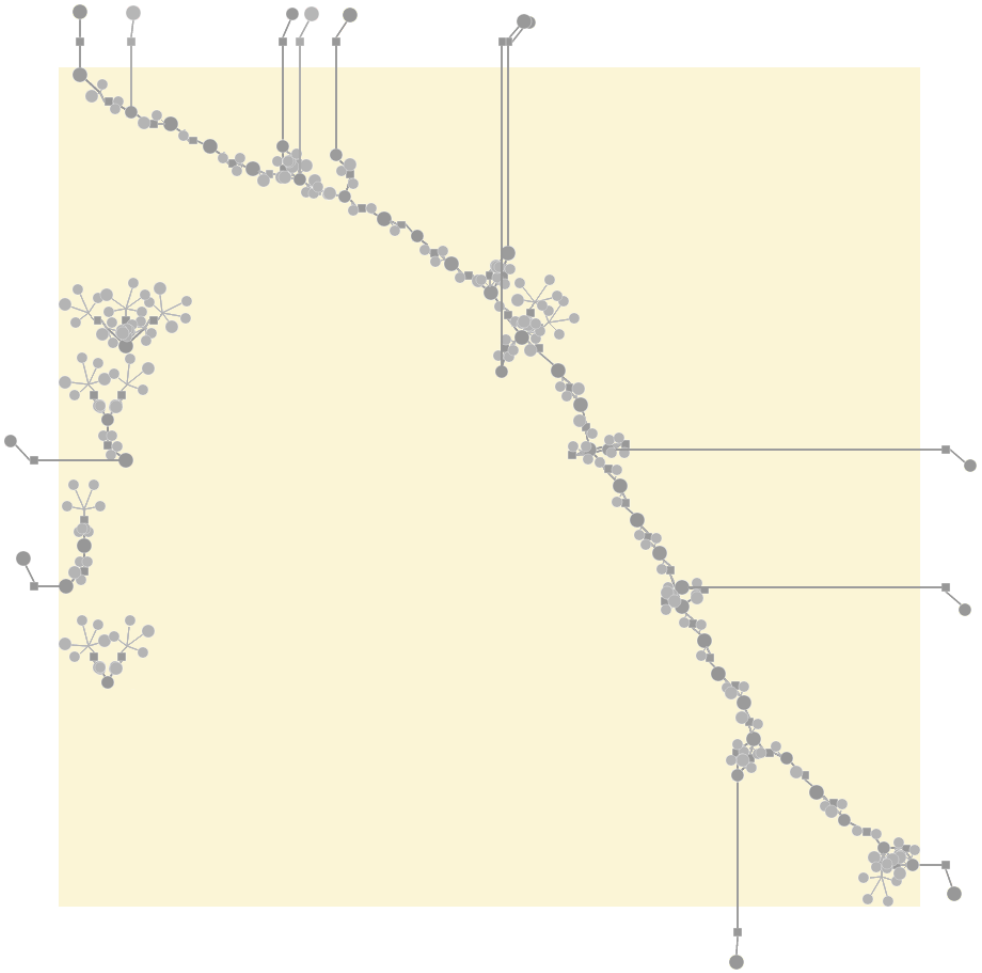

Fig. 1.2 **Sixty-seven reactions happening in the peroxisome compartment of the yeast *Y. lipolytica*** (*MODEL1111190000* [Loira et al., 2012]). Reactions are represented as squares linked by edges to their reactant and product metabolites (circles). The size of the figure does not allow for readable metabolite labels, so they are omitted. The reaction graph is disconnected as the transport reactions are not shown.

Much of the complexity of the reaction network comes from biochemically *similar reactions* that operate on slightly different substrates. For example, in the aforementioned peroxisome compartment of *Y. lipolytica* model six *acyl-CoA oxidase* reactions are present, transforming *fatty acyl-CoA*s differing in their carbon chain length (*decanoyl-CoA*, *lauroyl-CoA*, etc.) into the corresponding *unsaturated fatty acyl-CoA*s. These reactions correspond to the same Enzyme Commission number: EC 1.3.3.6. There are also several similar reactions for other steps of the *β-oxidation of fatty acids* pathway [Metzler

and Metzler, 2001]. Figure 1.3 shows the same processes as in Figure 1.2 but with similar metabolites and reactions colored accordingly. Grouping similar metabolites and similar reactions, would lead to a generalized peroxisome representation, as shown in Figure 1.4. The generalized model describes the *β-oxidation of fatty acids* pathway in a generic way: as a transformation of *saturated fatty acyl-CoA* into *fatty acyl-CoA (4-)*, then into *hydroxy fatty acyl-CoA*, *3-oxo fatty acyl-CoA*, and back to *saturated fatty acyl-CoA* (with a shorter carbon chain). The *beta-oxidation* chain of the reactions in the initial model, transforming step-by-step the *saturated fatty-acyl-CoA* with the longest carbon chain into the one with the shortest chain, in the generalized model appears as a cycle (generalizing all the *fatty-acyl-CoA*s into one metabolite, regardless the chain length).

Although all of these details are needed for accurate computer simulation, and are common to many models, it is often not them but instead the *differences from the common pattern* that demand curator's attention. These differences may be caused by errors in the model, such as missing steps or erroneous connections between pathways, or they may be organism-specific adaptations such as alternative pathways that are biologically interesting. For example, the generalized model of the peroxisome of *Y. lipolytica* (Figure 1.4) highlights the fact that there is a particularity concerning *C24:0-CoA (tetracosanoyl-CoA)* (red, inside the cycle): There exists a"short-cut" reaction, producing it directly from another *fatty acyl-CoA* (yellow), avoiding the usual four-reaction beta-oxidation chain, used for other *fatty acyl-CoA*s. An appropriate level of abstraction is needed to allow experts to explore and compare whole-genome networks.

1.7 Thesis aims and objectives

To this end, in this thesis we define a 3-level zoomable representation of metabolic models, that can be used by human experts during the curation and analysis step.

- The most abstract level represents *compartmentalization* of the model, and focuses on such questions as: Are all the compartments present? Are they well connected by transport reactions?

- The second level shows the *modules* inside of each of the compartments. The questions to be addressed on this level include: Are all the essential processes present? Is the structure of each process correct? Are there any organism-specific adaptations of the structure?

- The most detailed level is intended for computer simulation and represents the

inner structure of each of the modules with all the *metabolites, reactions* and their kinetics, stoichiometry and constraints.

The two abstract levels are intended for a human expert, and the last one for a computer.

We develop the algorithms for model generalization at the second level, and software exploiting this representation. In Figure 1.1, the processes marked with red arrows can potentially use this multilevel representation and model generalization to facilitate the human curators' work, and as a means of bringing several models to the same level of abstracting for their comparison or knowledge-based exploration.

1.8 Thesis overview

The rest of this thesis is organized as follows. To develop an understanding of the domain, main definitions and general introduction to metabolic modeling, related methods and software are given in Chapter 2.

In chapter 3 we formally define the method that we developed to detect similar metabolites and reactions in a metabolic network model. We also define the properties of generalized models, obtained by this method. Chapter 4 describes the applications of our model generalization method to 1 286 models from the Path2Model [Büchel et al., 2013] project, and demonstrates how the generalization helps to detect problems and particularities in metabolic networks.

Chapter 5 introduces a web-based system Mimoza that combines the model generalization method with the zoomable user interface techniques to create multilevel semantically zoomable representation of metabolic networks.

Finally, Chapter 6 summarizes the contributions of this thesis, and presents perspectives.

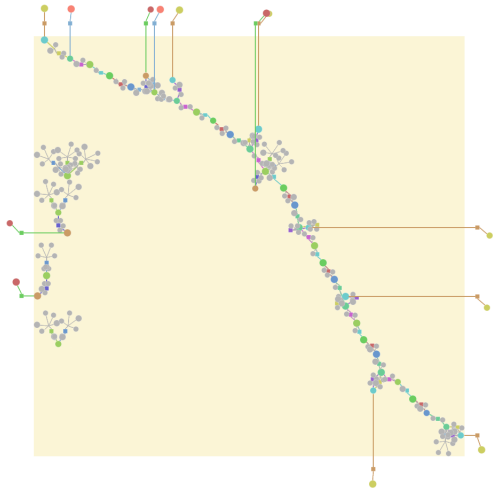

Fig. 1.3 **Sixty-seven reactions happening in the peroxisome compartment of the yeast *Y. lipolytica*** (*MODEL1111190000* [Loira et al., 2012]), with similar metabolites/reactions sharing the same color. The size of the figure does not allow for readable metabolite labels, so they are omitted. The reaction graph is disconnected as the transport reactions are not shown.

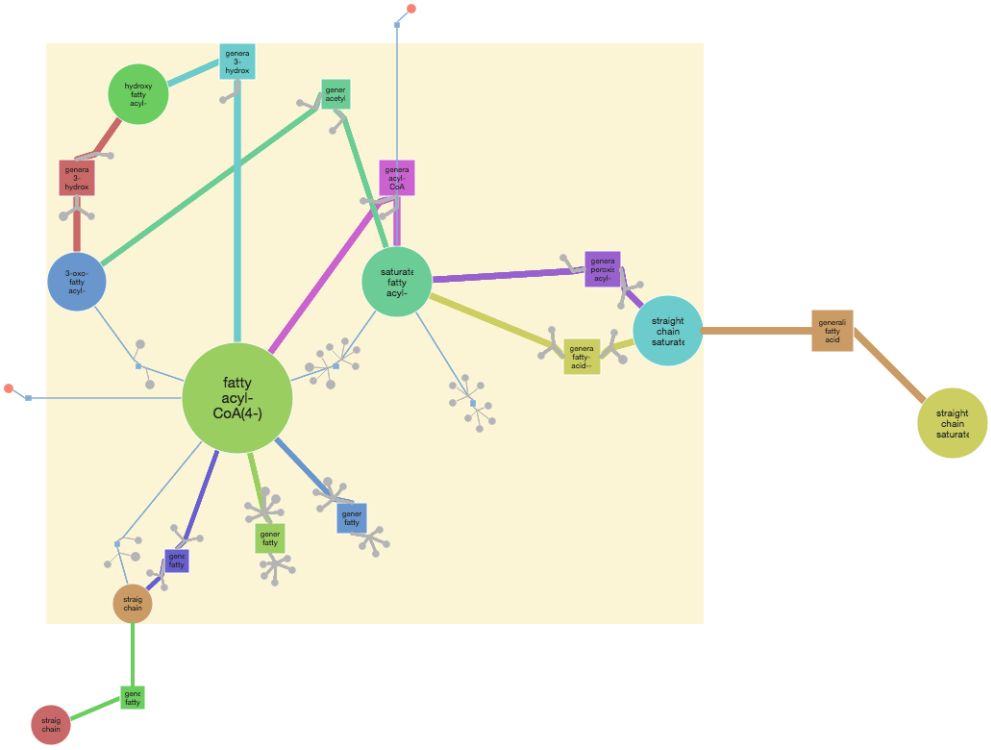

Fig. 1.4 **The generalized representation of the peroxisome compartment of the yeast *Y. lipolytica*** (*MODEL1111190000* [Loira et al., 2012]). Similar metabolites/reactions, that share the same color in Figure 1.3, are grouped into generalized metabolites/reactions, colored accordingly. The number given in parentheses and the size of each node indicates how many entities it generalizes.
Most of the disconnected reactions in Figure 1.3, for example, four *fatty acid oxidation* reactions (light blue), are reconnected to the main loop after generalization; highlighting the fact that they are part of the *β-oxidation of fatty acids* pathway.

Chapter 2

Background

This thesis builds on a great deal of existing work in metabolic modeling, knowledge representation, metabolic network reconstruction, and navigation in biological networks. These are each vast and widely studied subjects, and the literature is quite abundant. We introduce here the essential elements of this background information, focusing on those that are essential for the chapters that follow.

2.1 The organization of the cell

To develop methods that respect biological constraints as well as to provide usable tools to biologists for navigating the resulting networks, it is necessary to understand how the eukaryotic cell is organized. This organization imposes constraints on the way that reactions can be connected. It also provides a natural structuring of the network, that can be used for navigation.

The cell is the basic structural, functional and biological unit of all known living organisms, the "building block of life". There are two types of cells, *procaryotic* (microorganisms, bacteria, etc.) that are characterized by only one compartment, and *eukaryotic* that have the inner membrane that define the *nucleus*. The nucleus of the cell contains the genetic material or genome in form of the double-stranded DNA molecule. The area between the outer and inner membrane, including all of the components therein is called *cytoplasm* [Alberts et al., 2007].

In addition to the two main *compartments* (nucleus and cytoplasm), eucaryotic cells have *organelles,* that are smaller compartments with a membrane and which contain a set of specific enzymes. Material can be *transported* through the membranes directly or through gates.

The reason why we need to understand the organization of the cell in this work is that

it imposes the limits on the scope of model generalization. In order for generalization to be biologically sensible and mathematically possible, we perform it only inside compartments: similar metabolites can be grouped together only if they are located in the same compartment; same holds for reaction factoring. Compartments and their relative positions also define levels for navigation, starting from extracellular space and zooming into cytoplasm and organelles. Another important constraint for generalization procedure is the consistency of transport reactions: chemically equal metabolites that belong to different compartments should be generalized to the same level of abstraction, and the corresponding transport reactions should be factored together into generalized transport reactions.

2.2 Knowledge representations

To provide an additional semantic level, a model should be further enriched with the knowledge from biological databases and ontologies, by *annotation* of elements of the models (such as metabolites, reactions, compartment) with appropriate identifiers. Semantic information adds meaning to components of the model to help identify and interpret them unambiguously.

Ontologies are formal representations of knowledge with definitions of concepts, their attributes and relations between them expressed in terms of axioms in a well-defined logic [Rubin et al., 2008]. Ontologies also provide identifiers for the concepts that they describe, allowing to reference these concepts unambiguously.

Examples of the knowledge resources used to add a semantic level to metabolic models include ChEBI [de Matos et al., 2010], the database and ontology of Chemical Entities of Biological Interest. Among other entities, ChEBI describes small molecules (providing names, definitions, links to other databases, SMILES, InChI, their chemical roles, etc.) and relates them with each other (with hierarchical and other relationships, e.g., *decanoyl-CoA* is_a *medium-chain fatty acyl-CoA*). In metabolic models, metabolites are often annotated with their ChEBI identifiers.

Uniprot (Universal Protein Resource) [The UniProt Consortium, 2013] is a catalog of information on proteins, and can be used for annotation of enzymes, or of reactions catalyzed by those enzymes.

The Gene Ontology (GO) [Ashburner et al., 2000] provides controlled vocabularies of terms representing gene product properties. It consists of three main branches. The *cellular component* branch defines the parts of a cell and of extracellular environment, it can be used for compartment annotation. The *molecular function* defines the activities

that occur at the molecular level, e.g., binding or catalysis. Finally, the *biological process* branch defines molecular events pertinent to the functioning of cells, tissues, organs, and organisms. This branch can be used for annotation of reactions.

The Rhea manually annotated database of biochemical reactions [Alcántara et al., 2012] is a good source of knowledge for reaction annotation.

Kyoto Encyclopedia of Genes and Genomes (KEGG) [Kanehisa et al., 2012] provides a set of resources that can be used for model elements' annotation: reactions can be linked to KEGG Pathway entries, as well as KEGG Reaction and KEGG Orthology identifiers; metabolites can be annotated with KEGG Compound entries.

Elements of the model can be also annotated with the terms of the Evidence Ontology (ECO) [Chibucos et al., 2014], a controlled vocabulary that describes types of scientific evidence within the realm of biological research (such as laboratory experiments, computational methods, manual literature curation, and other means).

Three important ontologies in the field of systems biology are described in [Courtot et al., 2011]: the Systems Biology Ontology (SBO), the Kinetic Simulation Algorithm Ontology (KiSAO) and the Terminology for the Description of Dynamics (TEDDY). SBO defines the semantic information about model structure and its components. KiSAO is used to annotate the description of simulation experiments (e.g., encoded in Simulation Experiment Description Markup Language (SED-ML) [Köhn and Le Novère, 2008; Waltemath et al., 2011]) and supplies information about existing model simulation and analysis algorithms, and their interrelationships through their characteristics and parameters. TEDDY provides terms needed for description of numerical results: it classifies the temporal behaviors observed in a simulation, the diversifications and relationships between them, their characteristics, and the functional motifs generating particular types of behaviors.

These and other bio-ontologies can be found through the BioPortal [Whetzel et al., 2011] repository of biomedical ontologies, or the OBO Foundry [Smith et al., 2007].

To keep the representation of identifiers of ontological terms and knowledge-base entries unique and machine readable standardization efforts such as Identifiers.org [Juty et al., 2012] emerge.

In these thesis, we use the *cellular component* branch of GO to infer relative compartment positions (using *part_of* and *is_a* relationships defined between the compartments in GO) for correct model visualization. We use the hierarchical relationships defined in ChEBI for biologically sensible generalization of metabolites. While there exist various databases describing metabolites, ChEBI is the standard ontology in the biochemistry domain and defines not only the relevant terms but also the relationships between them.

2.3 Standards for conveying knowledge

The knowledge represented in a biological model must also be communicated, between software tools, and between software and users, in ways that preserve the semantics of the knowledge. Reliable communication of this knowledge is assured by international standards, that define *formats* and rules for interpreting them. Metabolic network models can be represented in various formats, depending on the purpose of the model: exchange between programs, or presentation to a human user.

2.3.1 Exchange formats

For instance, the Systems Biology Markup Language (SBML) [Hucka et al., 2003] is a free and open interchange XML-based format, widely adopted by the community. SBML is intended for computer models of biological processes and can be used for models of cell signalling, metabolism, gene regulation, etc. Various simulation and analysis tools accept models in SBML format,e.g., COPASI [Hoops et al., 2006], a software application for simulation and analysis of biochemical networks and their dynamics, COBRApy [Ebrahim et al., 2013], a toolbox for constraint-based reconstruction and analysis, FAME [Boele et al., 2012], a web-based flux analysis [Orth et al., 2010] and modeling environment, among many others.

The first SBML (level 1 version 1) specification [Hucka et al., 2001] was created in 2001 to provide a standard format for representation of the rapidly increasing number of models in systems biology. It described the format for representing the basic model structure: compartments, species (e.g., metabolites in the case of metabolic models), reactions (processes between those species), unit definitions, parameters and rules. Since then, a group of SBML editors and the community has been constantly working on improving the standard to address the growing needs of the modelers. For example, in level 2 version 1 [Finney and Hucka, 2003], additional model elements: events and function definitions were introduced.

Up to level 2 version 2, SBML was a syntax standard, which expresses the mathematical structure of models (i.e., the variables and their mathematical relationships), but does not define what those variables represent, nor how they were generated. In level 2 version 2 [Finney et al., 2006], a standard format for annotation of model elements with identifiers from various knowledge bases was presented, therefore allowing a modeler to provide an additional, semantic, level.

With the creation of SBML level 3 version 1 [Hucka et al., 2010] the core SBML that defines the general model structure was separated from the supplementary elements

specific to particular model types and purposes. Those supplementary elements were moved to dedicated model packages. Examples of packages include *annotations*, a package that supports richer annotation syntax than the regular annotations introduced in level 2 version 2, *flux balance constraints*, a package targeted to constraint-based metabolic models allowing to define information needed to perform the flux balance analysis (FBA) [Orth et al., 2010], *qualitative models*, a package for models wherein species do not represent quantity of matter and processes are not reactions per se, *layout*, a package that defines the spatial topology of a network diagram, *groups*, a package that provides a means of grouping model elements, etc.

CellML [Lloyd et al., 2004] is another XML-based format for storage and exchange of computer-based mathematical models. CellML includes information about model structure (relative organization of the model parts), mathematics (equations describing the underlying processes) and metadata (semantics). CellML describes the structure and underlying mathematics of cellular models in a very general way and has facilities for describing any associated metadata, while SBML is primarily aimed at exchanging information about pathway and reaction models. In CellML, the biological information is entirely stored in metadata rather than the language elements, like in SBML. Moreover, in SBML the mathematical expressions are more constrained than what is permitted in CellML.

BioPAX (Biological Pathway Exchange) [Demir et al., 2010] is a standard language to represent biological pathways at the molecular and cellular level. BioPAX can represent metabolic and signaling pathways, molecular and genetic interactions and gene regulation networks. BioPAX is defined in the Web Ontology Language (OWL) [McGuinness and van Harmelen, 2004] and is represented in the RDF/XML format. The scope of BioPAX is narrower than the one of SBML: SBML is meant to facilitate exchange and reuse of quantitative models, not necessarily limited to the biochemical pathways as BioPAX. BioPAX models cannot express information about sizes, amounts and kinetics, that can be contained in SBML model. But from the metadata point of view, BioPAX being an ontology, allows one to define the semantics of its elements in a richer way and more precisely than SBML.

2.3.2 Visualization formats

There are two packages developed for SBML level 3 that define the information needed for model visualization: *layout* and *render*. However, there exists a format especially targeted for model visualization: the Systems Biology Graphical Notation (SBGN) [Le Novère et al., 2009a]. It includes three orthogonal and complementary languages: the

Process Descriptions [Moodie et al., 2011], the Entity Relationships [Le Novere et al., 2011] and the Activity Flows [Mi et al., 2009].

The process description diagrams represent processes that convert physical entities into other entities, change their states or change their location. It is often used for the detailed drawing of metabolic networks. The entity relationship diagrams depict the interactions between entities and the rules that control them. Finally, the activity flows show the influence of biological activities on each other. They are very suitable for visualizing signaling pathways and gene regulatory networks.

In the SBGN Process Descriptions diagrams a metabolic model is represented as a graph: Reactions are visualized as square nodes connected by edges to round nodes representing their reactant and product metabolites. Figures 1.3 and 1.4 on page 11 show two networks presented in SBGN format.

In this thesis, we work with models in SBML format and use the layout package to store the model layout, and the groups package to represent the generalization of the model. For model visualization, we follow the SBGN Process Description language convention to choose the glyphs for model elements' representation: Metabolites are drawn as circles linked by edges to the reactions where they participate; reactions are represented as squares; compartments are drawn as rectangles. In Chapter 3 in order to define the model generalization procedure, we introduce a model representation as a pair of sets: metabolites and reactions.

2.4 Metabolic network reconstruction and transformation

So far we have seen representations of metabolic models, and formats for conveying them. In this section we present some background of tools that manipulate models through their representations. *Model reconstruction*, the word used in the literature, is in fact the inference of a new model from existing knowledge (gene annotations, existing models, reaction databases, etc.) and new knowledge obtained experimentally. *Model transformation* refers to modifying existing models, extending knowledge by deriving the consequences of existing knowledge, in a way that guarantees the consistency of the result.

Metabolic model reconstruction methods and tools are constantly becoming more and more advanced, and new ones are being developed. [Hamilton and Reed, 2014] provide a review of major software platforms for genome-scale metabolic network reconstruction. Model reconstruction tools semi-automatically create a draft model based

on genome data, using existing reaction and pathway databases, and models for similar organisms.

Among the existing methods, Pathway Tools [Karp et al., 2002] can be considered the *de facto* standard for *de novo* metabolic model reconstruction. The PathoLogic component of Pathway Tools takes an annotated genome in a Genbank [Benson et al., 2014] format as input, and produces a new pathway/genome database (PGDB) as output. It retrieves relevant reactions from the MetaCyc database [Caspi et al., 2012]. PathoLogic predicts the metabolic pathways of an organism and predicts what genes code for missing enzymes within the predicted pathways. The Pathway/Genome Editor components can be used after for curation of pathways, genes and enzymes in the newly created PGDB.

The RAVEN Toolbox [Agren et al., 2013] takes a genome for the species of interest and uses existing models for related organisms and/or the KEGG database, coupled with extensive gap-filling and quality control features, to provide a draft metabolic network reconstruction. It uses the protein homology to detect the conserved reactions.

The Model SEED [Devoid et al., 2013] creates a draft model from a genome sequence using the manually curated Model SEED database. It requires users to annotate their genome using RAST [Aziz et al., 2008], a fully-automated service for annotating bacterial and archaeal genomes.

The SuBliMinaL Toolbox merges reactions and pathway available for a given organism in KEGG and MetaCyc into a draft reconstruction. Existing metabolic models can also be incorporated into this process. The SuBliMinaL Toolbox is thus restricted to organisms found in those databases.

All of the aforementioned methods are limited to single-species reconstruction. Comparative ReConstruction (CoReCo) [Pitkänen et al., 2014] approach performs a simultaneous genome-scale metabolic reconstruction of multiple related species and leverages on the growing availability of sequenced genomes.

There also exist various software facilitating combining of existing models. Various challenges arise while merging SBML models. They include *syntactical requirements* (e.g., uniqueness of identifiers in the resulting model, no multiple assignments to variables, etc.), *semantical problems* (e.g., detection and merging of identical elements, detecting of biologically contradicting ones, such as overlapping compartments), and *loops of algebraic equations* that must be avoided.

One of the pioneering work in this area was SBMLmerge [Schulz et al., 2006]. It addresses the merging challenges through the use of four subroutines: SBMLannotate, SBMLcheck, SBMLmerge and SBML2dot. SBMLannotate assists the user in annotation of model elements, and searches for possible annotations in various knowledge bases, e.g.,

ChEBI, KEGG Compound, GO, etc. SBMLcheck performs various checks for model consistency: syntax check, annotation correctness and overlapping, consistency of mathematical rules, atom balancing in reactions. SBMLmerge combines the models, while detecting naming conflicts and conflicts between assignment rules. User is asked to solve those conflicts. SBML2dot plots the output model.

SemanticSBML [Krause et al., 2010] is a successor of SBMLmerge, it has focus on semantic annotations and in addition to sforementioned subroutines provides ones for calculating model difference, and for splitting SBML models.

Ontology Reasoning Engine for Molecular Pathways (OREMPdb) [Umeton et al., 2012] does not merge models into a new SBML model, but creates coherent ontologies out of different biochemical information sources. It consists of four modules: the *data access facility* extracts pathway information from existing biological databases, the *parser module* extracts relevant information from models in different formats (i.e., XML, RDF, SBML, CellML, etc.), the *core module* assembles this knowledge into a coherent ontology, finally, the *logic module* performs annotation of metabolites and runs automated comparison and identification of common metabolites and duplicate reactions. The duplicates are revealed to the user who should decide how to merge them.

PathCase-SB Model Composition Tool [Coskun et al., 2013] is another software for merging SBML models. It detects duplicated elements based either on user's input or on names and annotations of the elements in the case of automatic mode. The models to be merged should be compatible in terms of their SBML Levels. If the elements are not detected to be identical, both of them are added to the resulting model (which includes, for example, overlapping compartments).

The aforementioned model composition tools are powerful in automatic detection of common model elements based on their names and metadata, in well-annotated models. The automatic consistency checks are also well developed. However, the detected conflicts cannot be resolved automatically and require human expert's intervention.

In this thesis work we present another model transformation approach: the model generalization. It is completely automatic and does not require human intervention, the generalization is intrinsic to the models and is completely defined by its structure and metadata. Finally, the generalization is abstraction of the model which implies the loss of some of the details available in the initial model, even though the link between the initial model and the generalized one is preserved.

2.5 Navigation in biological networks

A metabolic network can be represented as a *bipartite* graph [Diestel, 2012] with two disjoint sets of nodes: metabolites and reactions, and edges that connect the reactions to their substrate and product metabolites. In SBGN format, the metabolite nodes are drawn as circles, and the reaction nodes as squares (figures 1.3 and 1.4).

Navigation in biological networks is essential to present the knowledge they contain in a way that helps the human user. As we have seen in section 1.5, exploration can aid in the interpretation of networks, but can also aid in the curation task. Mimoza (chapter 5) identifies and visualizes shortcuts and meanders in the network, that may be informative about errors in an inferred model, or about specifities of the modeled organism's metabolism that are revealed through the inferred model.

While the navigation in large graphs in general is beyond the scope of this thesis, the navigation in the large-scale metabolic network graphs remains a challenge, due to the complexity of those networks. Genome-scale metabolic models include thousands of reactions that may participate in organism's metabolism, e.g., 2 251 reactions in the metabolic network of the bacterium *Escherichia coli* [Orth et al., 2011], 2 352 reactions in the *yeast 7* metabolic network model of *Saccharomyces cerevisiae* [Aung et al., 2013], 7 440 reactions in *recon 2*, a global human metabolism reconstruction [Thiele et al., 2013]), while human experts understand best small-sized networks, containing up to hundreds of nodes [Herman et al., 2000; von Landesberger et al., 2011].

2.5.1 Desktop visualization tools

There exist various modeling tools for metabolic networks that also support visualization. Desktop tools include CellDesigner [Funahashi et al., 2008], VANTED [Rohn et al., 2012], and Cytoscape [Smoot et al., 2011]. They produce reasonably good visualizations of small networks (up to hundreds of reactions), but become cluttered at the genome-scale level, making the visualization unreadable.

2.5.2 Web-based visualization tools

Web-based tools allowing for metabolic network visualization are also emerging. JWS online [Snoep and Olivier, 2003], for example, provides a mechanism for network visualization using a force-directed layout algorithm [Fruchterman and Reingold, 1991; Tamassia, 2007]. It also encounters the aforementioned issues and thus is not capable of providing a readable representation for large networks.

MetDraw [Jensen and Papin, 2014] is an online tool for genome-scale metabolic model visualization, that makes use of decomposition of the model into compartments and pathways (if the pathway information is present in the model as a *subsystem* annotation of reactions) and duplication of minor metabolites. Metabolite duplication reduces clutter, but the huge number of reactions in the compartments of some models and missing *subsystem* annotations, makes the visualization consume too much space and do not allow a user to grasp the essential structure of the network.

2.5.3 Zooming user interfaces

Due to the huge numbers of reactions and of metabolites participating in multiple reactions, we have an uncomfortable choice between either many edge crossings in an automatic visualization of a genome-scale network, or over-duplication of various metabolites making the essential parts of the network disconnected and the visualization too large to grasp. Therefore an approach different to a simple graph layout algorithm is necessary. Zooming user interfaces (ZUI), which can change the size and nature of the content displayed at different zoom levels, provide a pertinent alternative. Two main types of magnification can be considered: *geometric zooming*, in which a region of the network is enlarged; and *semantic zooming*, in which additional properties are introduced with enlargement [Hu et al., 2007].

Semantic zooming was first introduced for biological data visualization in 1988 with Zomit [Pook et al., 1998], a generic application programming interface for developing servers for zoomable navigation and visualization, and illustrated with an example of ZoomMap, a prototype browser for HuGeMap human genome database [Barillot et al., 1998]. The work by Jianlu and Laidlaw [Jianu and Laidlaw, 2013] evaluates geometric zooming with the Google Maps interface on five examples (a gene co-regulation visualization, a gene expression heatmap viewer, a genome browser, a protein interaction network, and neural projections), and describes a positive feedback provided by both domain experts and less experienced users. Another example of a Google Maps-based ZUI is X:map [Yates et al., 2008], a genome annotation database that supports zoomable data browsing. It does not use semantic zooming, but allows for showing/hiding layers with additional information (EST and GenScan predictions).

There exist several web-based tools that include a zoomable representation of metabolic networks. Genome Projector [Arakawa et al., 2009] is a zoomable genome map with multiple views, including a pathway map. The pathway map is based on the Roche Biochemical Pathway wall chart available from the ExPASy proteomics server [Gasteiger et al., 2003]. The Roche Biochemical Pathway wall chart has a large size and shows the collec-

tion of biochemically known molecules, enzymes and reactions. Genome Projector provides a geometric zooming on the map and overlay layers to highlight reactions present in the organism of interest. The list of organisms is fixed to 320 bacterial genomes. The full Roche Biochemical Pathway map with the fixed layout is always shown, but only the reactions of interest (corresponding to the chosen organism) are highlighted.

NaviCell [Kuperstein et al., 2013] is a web environment that permits exploiting large maps of molecular interactions, including metabolic maps. It allows users to create their own maps, but does not provide a solution to the problem of huge network layout. The map creation is not fully automatic: The user must create a map in CellDesigner, export it as an image and partly manually edit it in a graphical designer to produce intermediate views (possibly with different level of details for semantic zooming). In addition, NaviCell permits a user to split the map into submaps called modules.

Another web-based tool, the Cellular Overview [Latendresse and Karp, 2011] creates interactive diagrams for metabolic maps of organisms in the BioCyc database [Caspi et al., 2012]. It is pathway-oriented, and supports only geometric zooming. Another drawback is that it does not show the compartmentalization.

The Reactome pathway database [Croft, 2013; Milacic et al., 2012] browser provides a zommable visualization of manually curated pathways for 19 organisms. It has two semantic zoom levels: a general representation of organism's pathways (nodes represent pathways, the edges connect the related ones); and submaps showing the details of each of the pathways, including compartmentalization. Several levels of geometric zoom are available on both semantic zoom levels. Reactome is pathway-oriented. Inside each pathway the layout is fixed: reactions, metabolites, and compartments common to two organisms have the same layout in corresponding representations. On the other hand, the positions and sizes of compartments might differ between pathways of the same organism.

None of the ZUI tools for metabolic map representation described above, except for NaviCell, allow users to input their own models. Moreover, as these examples show, not only geometric zoom but also model decomposition and semantic zoom are important for multi-level visualization of huge models. At the general level, the network needs to be decomposed into several meaningful modules (such as compartments, pathways). If after such a decomposition the model remains complicated (e.g. the mitochondrial compartment of the yeast consensus model [Herrgå rd et al., 2008] containing 230 reactions), a further decomposition is required.

We address these issues in Chapter 5 by introducing a model navigation tool Mimoza that combines the model generalization method and compartmentalization for model

decomposition with a ZUI.

Chapter 3

Knowledge-based generalization of metabolic models

3.1 Introduction

In this chapter we focus on the second level of abstraction of metabolic networks, that represents the modules inside compartments.

A fair amount of work has been done on identifying reusable modules. These approaches can be divided into two groups: *series* and *parallel*. A *series* approach operates on chains of reactions, and generalizes them as a series, consequently hiding the structure of the network. An example of a *series* approach is representing the network as a set of metabolic pathways (KEGG [Kanehisa et al., 2012], MetaCyC [Caspi et al., 2012]), that can be further divided, for example, into reaction modules (conserved sequences of reactions along the metabolic pathways) [Muto et al., 2013].

The other type of approach operates on reactions that are *parallel*, keeping the steps and preserving the general view of the network. An example of this approach is grouping reactions based on EC (Enzyme Commission) numbers [Tohsato et al., 2000]. The drawback of this approach is that it is not applicable to networks with no EC numbers assigned or reactions with no catalysing enzymes identified. We have developed another *parallel*-reaction method for knowledge-based generalization of metabolic models [Zhukova and Sherman, 2014a], which does not depend on enzyme information. It provides a higher-level view of a model while keeping its essential structure and omitting the details.

Definition 1 *The* model generalization *process groups metabolites present in the model into equivalence classes, and merges each class into a generalized metabolite. Reactions that involve same generalized metabolites are then factored together into a generalized*

reaction.

By applying the model generalization process, we can build a simplified model that focuses on the high-level relationships. The simplified model can be further divided into pathways.

3.2 Mathematical basis

3.2.1 Basic definitions

We represent a *metabolic model M* as a pair of two sets: a set S of metabolites, and a set R of reactions between them:

$$\begin{aligned} M &= \langle S, R\rangle && \text{- model,}\\ S &= \{s_1,\dots,s_n\} && \text{- metabolite set,}\\ R &= \{r_1,\dots,r_m\} && \text{- reaction set.} \end{aligned}$$

We represent each *reaction* $r \in R$ as a pair of sets of metabolites: its reactants and products. A chemical reaction may be represented by a balanced chemical equation, showing the formulae of the reactants and products, and the changes that take place [Clugston and Flemming, 2000]. This definition leads to restriction 3.1 that all the metabolites participating in the reaction must be different.

$$\begin{aligned} r = \quad & \langle \{s_1^{(rs)},\dots,s_k^{(rs)}\}, \{s_1^{(ps)},\dots,s_l^{(ps)}\}\rangle \in R \subset \langle 2^S \times 2^S\rangle,\\ & \text{where } s_1^{(rs)} \neq \dots \neq s_k^{(rs)} \neq s_1^{(ps)} \neq \dots \neq s_l^{(ps)} \end{aligned} \tag{3.1}$$

To perform the model generalization, we define an *equivalence operation* $\sim$ on the metabolite set, and group metabolites into equivalence classes: $[s]^{\sim} = \{\tilde{s} \in S | \tilde{s} \sim s\}$.

Metabolite equivalence imposes reaction equivalence: two reactions are equivalent if their corresponding reactant and product metabolite sets are pairwise equivalent.

$$\forall r, \tilde{r} \in R \begin{array}{l} r = \langle \{s_1^{(rs)},\dots,s_k^{(rs)}\}, \{s_1^{(ps)},\dots,s_l^{(ps)}\}\rangle,\\ \tilde{r} = \langle \{\tilde{s}_1^{(rs)},\dots,\tilde{s}_{\tilde{k}}^{(rs)}\}, \{\tilde{s}_1^{(ps)},\dots,\tilde{s}_{\tilde{l}}^{(ps)}\}\rangle \end{array}$$

$$r \sim \tilde{r} \iff \wedge \begin{cases} k = \tilde{k},\ l = \tilde{l} \\ \forall i \in \{0,\dots,k\}\ \exists!\, \tilde{i} \in \{0,\dots,\tilde{k}\} :\ s_i^{(rs)} \sim \tilde{s}_{\tilde{i}}^{(rs)} \\ \forall j \in \{0,\dots,l\}\ \exists!\, \tilde{j} \in \{0,\dots,\tilde{l}\} :\ s_j^{(ps)} \sim \tilde{s}_{\tilde{j}}^{(ps)} \end{cases}$$

Equivalent reactions are factored together into a generalized reaction that operates

on generalized metabolites (i.e., metabolite equivalence classes):

$$[r]^{\sim} = \langle \{[s_1^{(rs)}]^{\sim}, \ldots, [s_k^{(rs)}]^{\sim}\}, \{[s_1^{(ps)}]^{\sim}, \ldots, [s_l^{(ps)}]^{\sim}\} \rangle.$$

In order to maintain the number of distinct metabolites participating in a reaction, the *stoichiometry preserving restriction* 3.2, analogous to restriction 3.1, must be satisfied:

$$[s_1^{(rs)}]^{\sim} \neq \ldots \neq [s_k^{(rs)}]^{\sim} \neq [s_1^{(ps)}]^{\sim} \neq \ldots \neq [s_l^{(ps)}]^{\sim} \tag{3.2}$$

In order to avoid creation of paths in the generalized model that are not based on the evidence from the initial model, we introduce the *metabolite diversity restriction* 3.3: Metabolites that do not participate in any pair of equivalent reactions and do not have any common equivalent metabolites must not be grouped together:

$$\forall s \neq \tilde{s} \in S \; s \sim \tilde{s} \iff \vee \begin{cases} \exists r \neq \tilde{r} \in R : r \sim \tilde{r} \wedge s \in reactants(r) \wedge \tilde{s} \in reactants(\tilde{r}) \\ \exists r \neq \tilde{r} \in R : r \sim \tilde{r} \wedge s \in products(r) \wedge \tilde{s} \in products(\tilde{r}) \\ \exists \dot{s} \in S : \; s \tilde{\sim} \dot{s} \wedge \dot{s} \tilde{\sim} \tilde{s}. \end{cases} \tag{3.3}$$

Note that restriction 3.3 can be reformulated as maximizing the number of metabolite equivalence classes while keeping the reaction equivalence classes unchanged.

The *generalized model* $M/\sim$ is a pair of generalized metabolite and reaction sets (quotient sets):

$$\begin{array}{rcll} M/\sim & = & \langle S/\sim, R/\sim \rangle & \text{- generalized model,} \\ S/\sim & = & \{[s_1]^{\sim}, \ldots, [s_{\tilde{n}}]^{\sim}\} & \text{- quotient metabolite set,} \\ R/\sim & = & \{[r_1]^{\sim}, \ldots, [r_{\tilde{m}}]^{\sim}\} & \text{- quotient reaction set.} \end{array}$$

The generalized model is a *zoom out* of the initial model: It provides a higher-level view by including less metabolites and reactions, but more generic ones. For example, *3-oxodecanoyl-CoA*, *3-oxolauroyl-CoA*, and *3-oxohexanoyl-CoA* metabolites of the initial model can be generalized into *oxo-fatty acyl-CoA*.

Every reaction of the generalized model corresponds to at least one reaction of the initial model. This specific reaction has the same topology (numbers of distinct reactant and product metabolites) and operates on metabolites that can be zoomed out into those participating in the generalized reaction. An appropriate level of abstraction is defined with respect to the initial model as the most general one that satisfies restrictions 3.2 and 3.3.

The method and restrictions are described in figures 3.1-3.3.

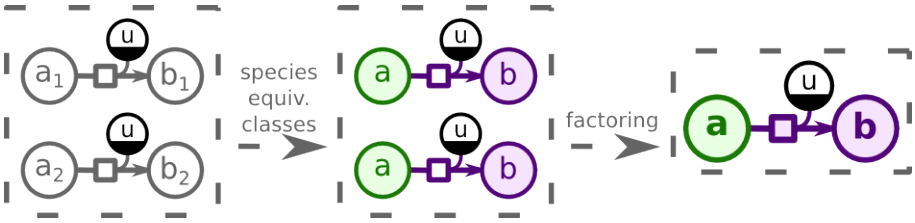

Fig. 3.1 **Model generalization method.** *Generalization* first groups the metabolites into equivalence classes, and then factors them into generalized metabolites. The reaction equivalence classes and factoring are inferred from the metabolite classes.

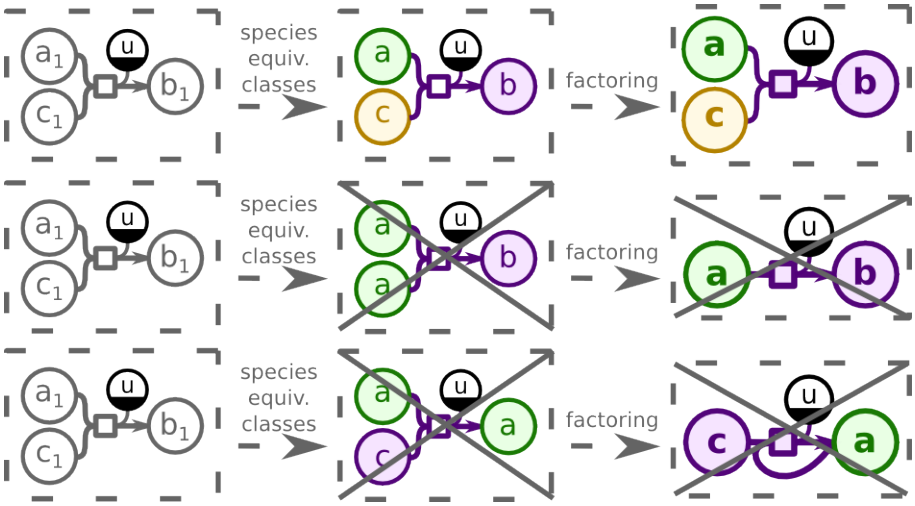

Fig. 3.2 **Stoichiometry preserving restriction.** The top part shows the correct generalization that obeys *restriction 3.2*. Two bottom parts show generalizations that would change the reaction stoichiometry, and thus are not allowed.

Specific and ubiquitous metabolites

We say that a *ubiquitous metabolite* is one that participates in many reactions (more than some threshold), such as *water*, *hydrogen*, *oxygen*, etc. Grouping of such metabolites would increase the number of reactions in which they participate even more. Besides that, these metabolites are already common to most of the models. In fact, during visualization ubiquitous metabolites are often even duplicated to improve readability [Rohn et al., 2012]. Consequently we do not generalize ubiquitous metabolites. In the generalized model each of them forms a trivial equivalence class:

$$S^{(ub)} = \{s_1^{(ub)}, \ldots, s_{\check{n}}^{(ub)}\} \subset S : \forall\, i\ [s_i^{(ub)}]^{\sim} = \{s_i^{ub}\}.$$

Specific metabolites are the others, which we divide into non-trivial equivalence classes and generalize accordingly.

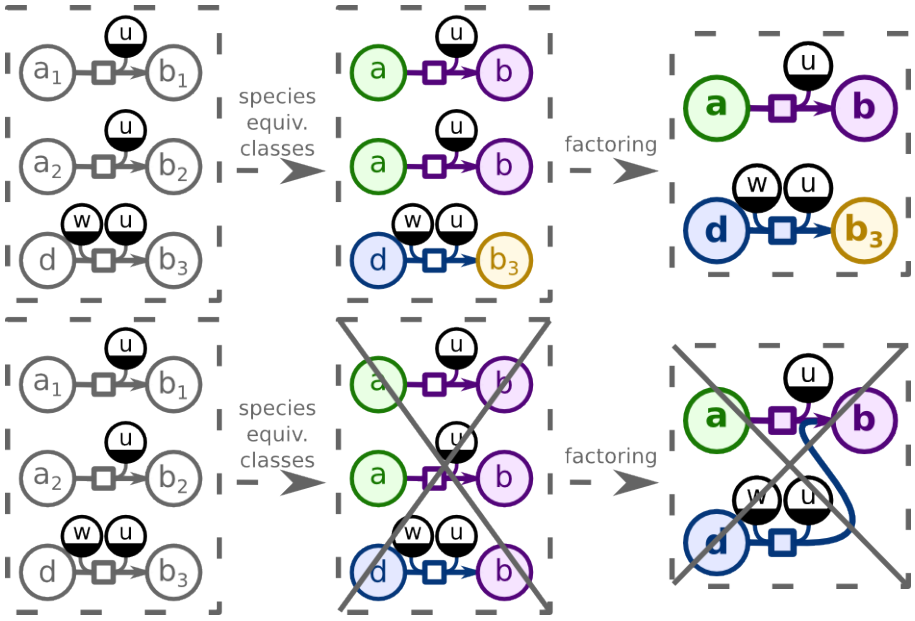

Fig. 3.3 **Metabolite diversity restriction.** The top part shows the correct generalization that obeys *restriction 3.3*. The bottom part violates the restriction as there is no evidence in the model (i.e., no equivalent reaction) of the metabolite b_3 belonging to the same equivalence class as b_1 and b_2.

3.2.2 Model generalization problem

Having agreed on terminology, we can now formally define the *model generalization problem*.

Problem 1 *Given a metabolic model* $M = \langle S, S^{(ub)} \subset S, R \rangle$ *that describes n metabolites (including* $\check{n} \leq n$ *ubiquitous ones) and m reactions, find an equivalence operation* $\sim$ *that obeys restrictions 3.2 and 3.3, and minimizes the number of reaction equivalence classes* $\sharp R/\sim$.

We will solve this problem in three steps:

1. Define the most general equivalence operation $\overset{\circ}{\sim}$ that corresponds to the minimal number of metabolite equivalence classes $\sharp S/\overset{\circ}{\sim}$, and does not take into account the restrictions;

2. Modify the current equivalence operation to satisfy the restriction 3.2;

3. Modify the current equivalence operation to satisfy the restriction 3.3.

3.2.2.1 Step 1. Equivalence operation $\mathring{\sim}$.

Definition 2 *Given a model* $M = \langle S, S^{(ub)} \subset S, R\rangle : \sharp S = n, \sharp S^{(ub)} = \check{n} \leq n, \sharp R = m$*, we define an* equivalence operation $\mathring{\sim}$ *on the metabolite set* S *as forming* $\check{n}+1$ *equivalence classes in the quotient set* $S/\mathring{\sim}$*: one for each of the ubiquitous metabolites, and one for all the other metabolites:*

$$\begin{aligned} &\forall\, s^{(ub)} \in S^{(ub)} \quad [s^{(ub)}]^{\mathring{\sim}} = \{s^{(ub)}\}, \\ &\forall\, s, \tilde{s} \in S \backslash S^{(ub)} \quad [s]^{\mathring{\sim}} = [\tilde{s}]^{\mathring{\sim}} = S \backslash S^{(ub)}. \end{aligned}$$

Lemma 1 *For any equivalence operation* $\sim$ *on the model* $M = \langle S, S^{(ub)} \subset S, R\rangle$*, the corresponding quotient metabolite set* $S/\sim$ *and quotient reaction set* $R/\sim$ *are partitions of, respectively, the quotient metabolite set* $S/\mathring{\sim}$ *and the quotient reaction set* $R/\mathring{\sim}$ *induced by* $\mathring{\sim}$*:*

$$\forall \text{ equivalence operation } \sim \text{ defined on } \langle S, S^{(ub)}, R\rangle : \wedge \begin{cases} \forall\, s \in S & [s]^{\sim} \subset [s]^{\mathring{\sim}}, \\ \forall\, r \in R & [r]^{\sim} \subset [r]^{\mathring{\sim}}. \end{cases}$$

To build the quotient metabolite and reaction sets induced by the equivalence operation $\mathring{\sim}$ we use Algorithm 1 that forms equivalence classes for ubiquitous and then specific metabolites as in Definition 2 and then computes generalized reactions.

3.2.2.2 Step 2. Stoichiometry preserving restriction

Problem 2 *Given an equivalence operation* $\sim$ *defined on a metabolic model* $M = \langle S, S^{(ub)} \subset S, R\rangle$ *find an equivalence operation* $\check{\sim}$ *that obeys restriction 3.2 and induces a quotient metabolite set* $S/\check{\sim}$ *of minimal size* $\sharp S/\check{\sim}$*, such that* $S/\check{\sim}$ *is a partition of the quotient metabolite set* $S/\sim$ *induced by* $\sim$*, i.e.,* $\forall s \in S\ [s]^{\check{\sim}} \subset [s]^{\sim}$.

To satisfy restriction 3.2 we start with the given equivalence operation $\sim^0 = \sim$, and iteratively improve it, until the stoichiometry preserving property 3.2 is obeyed (see Algorithm 2). We denote the equivalence operation obtained at the i-th iteration step as $\sim^i$.

At each iteration, if there exists a metabolite equivalence class that violates the stoichiometry preserving property 3.2, i.e.,:

$$\exists\, s \neq \tilde{s} \in S, r \in R : s \in metabolites(r) \wedge \tilde{s} \in metabolites(r) \wedge [s]^{\sim^i} = [\tilde{s}]^{\sim^i},$$

we partition this metabolite equivalence class $[s]^{\sim^i} = [\tilde{s}]^{\sim^i}$ into two: $[s]^{\sim^{i+1}} \vee [\tilde{s}]^{\sim^{i+1}} = [s]^{\sim^i} = [\tilde{s}]^{\sim^i}$ to form a new approximation $\sim^{i+1}$ of the equivalence operation. When no metabolite equivalence class that violates the restriction 3.2 can be found, the current equivalence operation is returned as the result.

Algorithm 1: Compute$\overset{\circ}{\sim}$

Data: $M = \langle S, S^{(ub)} \subset S, R \rangle : \sharp S = n, \sharp S^{(ub)} = \check{n} \leq n, \sharp R = m$ - metabolic model describing n metabolites, $\check{n}$ among them being ubiquitous, and m reactions.

Result: $\overset{\circ}{\sim}$ - equivalence operation described in Lemma 1, $M/\overset{\circ}{\sim} = \langle S/\overset{\circ}{\sim}, S^{(ub)}/\overset{\circ}{\sim} \subset S/\overset{\circ}{\sim}, R/\overset{\circ}{\sim} \rangle$ - corresponding generalized model.

$S/\overset{\circ}{\sim} \leftarrow \emptyset$ // resultant quotient metabolite set $S/\overset{\circ}{\sim} \subset 2^S$

$S^{(ub)}/\overset{\circ}{\sim} \leftarrow \emptyset$ // res. quotient ubiq. metabolite set $S^{(ub)}/\overset{\circ}{\sim} \subset 2^{S^{(ub)}}$

$R/\overset{\circ}{\sim} \leftarrow \emptyset$ // resultant quotient reaction set $R/\overset{\circ}{\sim} \subset 2^R$

$\overset{\circ}{\sim} \leftarrow \emptyset$ // resultant equivalence operation $\overset{\circ}{\sim} : S \cup R \rightarrow S/\overset{\circ}{\sim} \cup R/\overset{\circ}{\sim}$

/* Generalize ubiquitous metabolites */

for $s^{(ub)} \in S^{(ub)}$ **do**

  $[s^{(ub)}]^{\overset{\circ}{\sim}} \leftarrow \{s^{(ub)}\}$ // map $s^{(ub)}$ to its equivalence class

end for

$S^{(ub)}/\overset{\circ}{\sim} \leftarrow \{[s^{(ub)}]^{\overset{\circ}{\sim}} | s^{(ub)} \in S^{(ub)}\}$

/* Generalize specific metabolites */

for $s \in S \backslash S^{(ub)}$ **do**

  $[s]^{\overset{\circ}{\sim}} \leftarrow S \backslash S^{(ub)}$ // map s to its equivalence class

end for

$S/\overset{\circ}{\sim} \leftarrow S^{(ub)}/\overset{\circ}{\sim} \cup \{S \backslash S^{(ub)}\}$

/* Generalize reactions */

// map a reaction to its generalized version

$gen \leftarrow \lambda r. \langle \{[s]^{\overset{\circ}{\sim}} | s \in reactants(r)\}, \{[s]^{\overset{\circ}{\sim}} | s \in products(r)\} \rangle$

for $r \in R$ **do**

  $[r]^{\overset{\circ}{\sim}} \leftarrow \{\tilde{r} \in R | gen(\tilde{r}) = gen(r)\}$

end for

$R/\overset{\circ}{\sim} \leftarrow \{[r]^{\overset{\circ}{\sim}} | r \in R\}$

return $\overset{\circ}{\sim}, \langle S/\overset{\circ}{\sim}, S^{(ub)}/\overset{\circ}{\sim}, R/\overset{\circ}{\sim} \rangle$

At each iteration one equivalence metabolite class is partitioned. In the worst case, the equality operation = (each metabolite is equivalent only to itself) will be achieved. As it obeys restriction 3.2, the process will terminate.

Algorithm 2: PreserveStoichiometry

Data: $\sim$ - equivalence operation defined on a metabolic model
$M = \langle S, S^{(ub)} \subset S, R \rangle$, $M/\sim = \langle S/\sim, S^{(ub)}/\sim \subset S/\sim, R/\sim \rangle$ - corresponding generalized model.

Result: $\tilde{\sim}$ - equivalence operation described in Problem 2,
$M/\tilde{\sim} = \langle S/\tilde{\sim}, S^{(ub)}/\tilde{\sim} \subset S/\tilde{\sim}, R/\tilde{\sim} \rangle$ - corresponding generalized model.

$S/\tilde{\sim} \leftarrow S/\sim$ // resultant quotient metabolite set $S/\tilde{\sim} \subset 2^S$
$S^{(ub)}/\tilde{\sim} \leftarrow S^{(ub)}/\sim$ // res. q. ubiq. metabolite set $S^{(ub)}/\tilde{\sim} \subset 2^{S^{(ub)}}$
$R/\tilde{\sim} \leftarrow \emptyset$ // resultant quotient reaction set $R/\tilde{\sim} \subset 2^R$
$\tilde{\sim} \leftarrow \sim$ // resultant equivalence operation $\tilde{\sim} : S \cup R \rightarrow S/\tilde{\sim} \cup R/\tilde{\sim}$

/* Partition quotient metabolites to obey restriction 3.2 */
for $S^{(gen)} \in \{\tilde{S}^{(gen)} \in S/\tilde{\sim} | \exists s \neq \tilde{s} \in \tilde{S}^{(gen)}, r \in R : s \in metabolites(r) \wedge \tilde{s} \in metabolites(r)\}$ **do**
  $\Pi = Partition(S^{(gen)})$
  $S/\tilde{\sim} \leftarrow \Pi \cup S/\tilde{\sim} \setminus \{S^{(gen)}\}$ // Update $S/\tilde{\sim}$
  for $\tilde{S}^{(gen)} \in \Pi$ **do**
    for $s \in \tilde{S}^{(gen)}$ **do**
      $[s]^{\tilde{\sim}} \leftarrow \tilde{S}^{(gen)}$ // Update $\tilde{\sim}$
    end for
  end for
end for

/* Generalize reactions */
// map a reaction to its generalized version
$gen \leftarrow \lambda r. \langle \{[s]^{\tilde{\sim}} | s \in reactants(r)\}, \{[s]^{\tilde{\sim}} | s \in products(r)\} \rangle$
for $r \in R$ **do**
  $[r]^{\tilde{\sim}} \leftarrow \{\tilde{r} \in R | gen(\tilde{r}) = gen(r)\}$
end for
$R/\tilde{\sim} \leftarrow \{\tilde{\sim}(r) | r \in R\}$

return $\tilde{\sim}, \langle S/\tilde{\sim}, S^{(ub)}/\tilde{\sim}, R/\tilde{\sim} \rangle$

Metabolite equivalence class partition

Clique partition

Definition 3 *For a given a set of metabolites and a set of reactions between them, we define a* metabolite compatibility graph *as a simple undirected graph with vertices representing*

the metabolites, and edges linking those of the metabolites that do not participate in the same reaction (i.e., putting them into the same equivalence class does not violate the stoichiometry preserving restriction 3.2).

Note, that any set of metabolites that can be put into the same equivalence class without violating the restriction 3.2, forms a clique in the metabolite compatibility graph, i.e., a complete subgraph: for every pair of its vertices there exists an edge linking them. Thus, the problem of partition the metabolite equivalence class into minimum number of classes, such that all of them obey the restriction 3.2 is a clique partition problem.

Problem 3 (Clique partition) *Find the smallest number of cliques in a graph such that every vertex in the graph is represented in exactly one clique.*

Remark 1 *Clique partition problem is known to be NP-complete [Bhasker and Samad, 1991].*

In a metabolite compatibility graph, there are usually a few edges missing, i.e., in each metabolite equivalence class that violates the restriction 3.2 there are usually only a few conflicts present, and multiple solutions of the partition problem exist.

Metabolite ontology. In order to make the choice of the metabolite equivalence classes biologically meaningful, we use an ontology that describes hierarchical *is_a* relationships (more specific to more general) between metabolites.

Definition 4 *A term t is a* model term *if it corresponds to a specific metabolite in the metabolic model.*

We assume that no two model terms are connected by a descendant-ancestor (more specific–more general) relationship in the ontology; otherwise, we mark the ancestor term ubiquitous:

$$\forall\, t, T \in terms: \left(\begin{array}{c} \exists metabolites(t) \in S \\ \wedge\ \exists metabolites(T) \in S \\ t \in descendants(T) \end{array} \right) \Rightarrow t = T.$$

We iteratively remove all the leaf terms that are not model terms from the ontology, so that all the model terms become leaves, and all the leaves become model terms.

For each metabolite equivalence class that needs to be partitioned, we first find the least common ancestor T of the ontological terms corresponding to its metabolites. If the ontology allows for multiple inheritance, and there are several such least common

ancestors, we pick a random one. Then we look among the T-th descendant terms for those that are compatible (to avoid multiple inheritance).

Definition 5 *Terms $t_1, \ldots, t_k$ are* compatible *if and only if their descendant model terms do not intersect:*

$$t_1, \ldots, t_k \text{ are compatible} \iff \forall i \neq j \in \{1, \ldots, k\}\ descendants(t_i) \cap descendants(t_k) = \emptyset.$$

Problem 4 *Given a term T, find a compatible term set among its descendants, such that it has minimal size, covers all the T-th descendant leaf terms, and satisfies the stoichiometry preserving property 3.4:*

$$? \ t_1, \ldots, t_k \in descendants(T) : \wedge \begin{cases} k = k_{min}, \\ t_1, \ldots, t_k \text{ are compatible}, \\ leaves(T) \subset descendants(t_1) \cup \ldots \cup descendants(t_k), \\ \forall i \in \{1, \ldots, k\}, \forall r \in R : \\ \quad \sharp(metabolites(leaves(t_i)) \cap metabolites(r)) \leq 1. \end{cases} \tag{3.4}$$

To do so, we first exclude all the terms that violate the stoichiometry preserving property 3.4. We thus obtain an exact set cover problem.

Problem 5 (Set cover) *Given a set X and a collection of its finite subsets Ψ, such that $\bigcup_{S \in \Psi} S = X$, find a minimum-size subset $\Pi \subset \Psi$ whose members cover all of X: $\bigcup_{S \in \Pi} S = \bigcup_{S \in \Psi} S = X$.*

Remark 2 *Set cover is NP-complete [Karp, 1972].*

Problem 6 (Exact set cover) *As in* Set cover *problem, except that here the sets used in the cover are not allowed to intersect.*

Remark 3 *Exact cover is NP-complete [Goldreich, 2008].*

Exact set cover applied to ontological terms. Each ontological term t defines a set $S(t)$ of its descendant leaf terms (including t if it is a leaf). The instance consists of a set X of the model terms of interest, and a collection Ψ of all sets defined by their common ancestor T, its descendant terms, and their relative complements with respect to X: $\forall S \in \Psi\ X \setminus S \in \Psi$, excluding all the sets that violate the stoichiometry preserving property 3.4. We look for a minimal-size exact cover of X.

Note, that in this case an exact cover always exists, e.g., the one formed by all the leaf terms.

Choice of the ontology. We assume that any term that violates property 3.4 is removed from the ontology. Note that the term T is also removed.

If the ontology has no multiple inheritance, i.e., $\forall S, \tilde{S} \in \Psi \; S \cap \tilde{S} \neq \emptyset \Rightarrow S \subseteq \tilde{S} \vee \tilde{S} \subseteq S$, the problem becomes trivial: the set of the root terms forms the solution. The size of the solution, though, depends on the characteristics of the ontology, e.g., for a completely flat ontology (i.e., with no relationships) the solution consists of singleton equivalence classes.

If multiple inheritance is allowed, any $\Psi \subseteq 2^X$ becomes possible, and the problem becomes *NP*-complete.

We use the ChEBI ontology [de Matos et al., 2010] of chemical compounds, as it is *de facto* a standard for metabolite annotation in metabolic models. ChEBI consists of three main branches: *chemical entity*, *role*, and *subatomic particle*. The *chemical entity* branch describes terms useful for annotation of metabolites in a metabolic model. As of ChEBI version 101, this branch contains 37 693 terms, among which 29 888 are leaves. ChEBI has multiple inheritance with average number of parents 1.4 per term. Average number of siblings is also 1.4 per term. Maximal depth in the *chemical entity* branch is 28, while the average one is 11.

The level of detail in the ChEBI hierarchy is not uniform: some sub-branches are more developed than others, so equally precise terms may be placed unequally deep in the hierarchical tree. For example, both *hydrogen peroxide (CHEBI:16240)* and *decanoyl-CoA (CHEBI:28493)* terms describe precise chemical molecules; but *hydrogen peroxide* is only 5 terms away from the *chemical entity* in the ChEBI hierarchy, while *decanoyl-CoA* is 11 terms away.

Besides that, different types of classification are combined together in the hierarchical tree, leading to multiple inheritance. For example, in the *fatty acid (CHEBI:35366)* sub-branch, several classification types are present, including:

- classification based on the length of the carbon chain:
  - *short-chain fatty acid (CHEBI:26666)*: 2-4 carbons;
  - *medium-chain fatty acid (CHEBI:59554)*: 6-12 carbons;
  - *long-chain fatty acid (CHEBI:15904)*: 14-22 carbons;
  - *very long-chain fatty acid (CHEBI:27283)*: 24 -26 carbons;

- classification based on the presence of double bonds in the carbon chain:
  - *saturated fatty acid (CHEBI:26607)*: no double bonds;
  - *unsaturated fatty acid (CHEBI:27208)*: one or more double bonds;
- classification based on substituent groups:
  - *hydroxy fatty acid (CHEBI:24654)*: one or more hydroxy substituents;
  - *oxo fatty acid (CHEBI:59644)*: at least one aldehydic or ketonic group;
  - etc.

Moreover, using only hierarchical relationships in the ChEBI ontology is not always enough. Examples show, that similar reactions can happen to the acid and the base in a conjugate acid-base pair. A *conjugate acid-base pair* is two metabolites, one an acid and one a base, that differ from each other through the loss or gain of a proton [Stoker, 2012]. For instance, in the Rhea database of chemical reactions [Alcántara et al., 2012], the *acyl-CoA oxidase (RHEA:28354)* reaction: *decanoyl-CoA+FAD+H+* $\rightarrow$ *trans-dec-2-enoyl-CoA+*$FADH_2$ is found for both *decanoyl-CoA (CHEBI:28493)* and its conjugate base *decanoyl-CoA(4-) (CHEBI:61430)*. But hierarchically these metabolites are very far from each other in the ChEBI ontology: Their least common ancestor is *molecular entity (CHEBI:23367)*, a direct descendant of the root *chemical entity*. To establish a conjugate acid-base pair correspondence in the ChEBI ontology, not the hierarchical (*is_a*) but the special *is_conjugate_base_of* / *is_conjugate_acid_of* relationships are used. To maximize the chances of a conjugate acid-base pair being in the same quotient metabolite set, we generalize the hierarchical relationship.

Definition 6 *Term t is a* generalized direct descendant/ancestor *of a term T if and only if t or a conjugate base or acid of t is a direct descendant/ancestor of T or of a conjugate base or acid of T.*

Definition 7 *Term t is a* generalized descendant/ancestor *of a term T if and only if t is a generalized direct descendant/ancestor of T or of any generalized descendant/ancestor of T.*

We extend Ψ so that it is closed under the operation of relative complement: $\forall S, \tilde{S} \in \Psi\ S \setminus \tilde{S} \in \Psi$. This allows for solving the set cover problem instead of the exact cover one: As Ψ is closed under the operation of complement intersection, we can obtain an exact set cover $\tilde{C}$ from any set cover $C = \{S_1, S_2, \ldots, S_m\}$ by replacing its elements with their relative complements with the previous elements of C: $\tilde{C} = \{S_1, S_2 \setminus S_1, \ldots, S_m \setminus \bigcup_{i=1}^{m-1} S_i\}$.

Greedy set cover algorithm. To approximate the solution of the set cover problem, we use a greedy algorithm (see Algorithm 3): Among the available subset candidates $S_i \in \Psi$, pick the one of the largest size and add it to the resulting set cover Π. Repeat this operation until all elements of X are covered.

Algorithm 3: GreedySetCover

Data: X - set of interest, $\Psi \subseteq 2^X$ - set of subsets of X
Result: $\Pi \subseteq \Psi$ - set cover of X

$\Pi \leftarrow \emptyset$ `// resultant cover`

while $X \neq \emptyset$ **do**
  `//` $S \in \Psi$ `that covers maximum elements of` X
  $S^{(max)} \leftarrow max(\Psi, criterion = \lambda S.\sharp(S \cap X))$
  $\Psi \leftarrow \Psi \backslash \{S^{(max)}\}$
  $X \leftarrow X \backslash S^{(max)}$
  $\Pi \leftarrow \Pi \cup \{S^{(max)}\}$
end while

return Π

Greedy set cover is a polynomial time approximation algorithm that achieves an approximation ratio of $H(\sharp X)$, where $H(n)$ is the n-th harmonic number: $H(n) = \sum_{i=1}^{n} \frac{1}{i} \leq \ln n + 1$ [Chvatal, 1979]. It is the best possible polynomial time approximation algorithm for set cover, under plausible complexity assumptions [Feige, 1998].

3.2.2.3 Step 3. Metabolite diversity restriction

Problem 7 *Given an equivalence operation* $\sim$ *defined on a metabolic model* $M = \langle S, S^{(ub)} \subset S, R \rangle$*, find an equivalence operation* $\tilde{\sim}$ *that obeys restriction 3.3 and does not change the reaction equivalence classes:* $R/\sim = R/\tilde{\sim}$.

To satisfy restriction 3.3 we first associate each metabolite s in the initial model to a pair of sets of reaction equivalence classes in the quotient reaction set $R/\sim$, induced by reactions where it participates as a reactant or product:

$$s \rightarrow \langle R_s^{(rs)} = \{[r_1^{(rs)}]^{\sim}, \ldots, [r_o^{(rs)}]^{\sim}\}, R_s^{(ps)} = \{[r_1^{(ps)}]^{\sim}, \ldots, [r_t^{(ps)}]^{\sim}\} \rangle.$$

We then define an *equivalence operation* $\tilde{\sim}$ as forming a separate metabolite equivalence class for each of the ubiquitous metabolites, and putting $\sim$-equivalent specific metabolites that intersect in their product or reactant reaction classes in the same equivalence class:

$$\begin{aligned} \forall\, s^{(ub)} \in S^{(ub)}\ \forall\, s \in S \quad s^{(ub)} \tilde{\sim} s \quad &\Longleftrightarrow \quad s^{(ub)} = s, \\ \forall\, s, \tilde{s} \in S \backslash S^{(ub)} \qquad s \tilde{\sim} \tilde{s} \quad &\Longleftrightarrow \quad \wedge \begin{cases} s \sim \tilde{s}, \\ \vee \begin{cases} R_s^{(rs)} \cap R_{\tilde{s}}^{(rs)} \neq \emptyset \\ R_s^{(ps)} \cap R_{\tilde{s}}^{(ps)} \neq \emptyset \\ \exists\, \dot{s} \in S:\ s \tilde{\sim} \dot{s} \wedge \dot{s} \tilde{\sim} \tilde{s} \end{cases} \end{cases} . \end{aligned}$$

These steps are listed in Algorithm 4.

Any further partition of the quotient metabolite set would imply the partition of the quotient reaction set. Hence the number of metabolite equivalence classes is maximal for the current number of reaction equivalence classes, and restriction 3.3 is satisfied.

3.2.2.4 Complete algorithm

The complete algorithm starts with the aggressive metabolite and reaction groupings defined by the equivalence operation $\mathring{\sim}$ (see Definition 2), then ungroups some of metabolites and reactions to satisfy the stoichiometry preserving property 3.2, and, finally, ungroups some metabolites to satisfy the metabolite diversity property 3.3. For further details, see Algorithm 5.

3.3 Discussion

We have developed a method that provides a semantically zoomed-out view of a metabolic model, that keeps its essential structure but hides the details.

We have implemented our method as a Python program, that is available for download from http://metamogen.gforge.inria.fr. It takes a model in SBML format as an input, annotates its metabolites with ChEBI terms (if the annotations are not present in the model) and generalizes it. It produces two SBML files as an output. The first output file contains the generalized model. The second output file uses the groups extension [Hucka, 2012] of SBML, and contains the initial model plus a group that represents ubiquitous metabolites and groups for all non-trivial quotient metabolite and reaction sets (see Figure 3.4).

Algorithm 4: Maximize

Data: $\sim$ - equivalence operation defined on a metabolic model
$M = \langle S, S^{(ub)} \subset S, R \rangle$, $M/\sim = \langle S/\sim, S^{(ub)}/\sim \subset S/\sim, R/\sim \rangle$ - corresponding generalized model.

Result: $\tilde{\sim}$ - equivalence operation described in Problem 7,
$M/\tilde{\sim} = \langle S/\tilde{\sim}, S^{(ub)}/\tilde{\sim} \subset S/\tilde{\sim}, R/\tilde{\sim} \rangle$ - corresponding generalized model.

$S/\tilde{\sim} \leftarrow \emptyset$ // resultant quotient metabolite set $S/\tilde{\sim} \subset 2^S$

$S^{(ub)}/\tilde{\sim} \leftarrow S^{(ub)}/\sim$ // res. q. ubiq. metabolite set $S^{(ub)}/\tilde{\sim} \subset 2^{S^{(ub)}}$

$R/\tilde{\sim} \leftarrow R/\sim$ // resultant quotient reaction set $R/\tilde{\sim} \subset 2^R$

$\tilde{\sim} \leftarrow \sim$ // resultant equivalence operation $\tilde{\sim} : S \cup R \rightarrow S/\tilde{\sim} \cup R/\tilde{\sim}$

/* Update specific metabolite generalization */

// Map a metabolite to a set of its ~-equivalent metabolites

// that participate in ~-equivalent reactions

$r\_sim \leftarrow \lambda s.\{\tilde{s} \sim s | \exists r, \tilde{r} \in R : s \in reactants(r) \wedge \tilde{s} \in reactants(\tilde{r}) \wedge r \sim \tilde{r}\}$

$p\_sim \leftarrow \lambda s.\{\tilde{s} \sim s | \exists r, \tilde{r} \in R : s \in products(r) \wedge \tilde{s} \in products(\tilde{r}) \wedge r \sim \tilde{r}\}$

$sim \leftarrow \lambda s.r\_sim(s) \cup p\_sim(s)$

$S/\tilde{\sim} \leftarrow S^{(ub)}/\tilde{\sim} \cup \{sim(s) | s \in S \backslash S^{(ub)}\}$

// Merge all quotient metabolite sets that intersect

while $\exists S^{(gen)} \neq \tilde{S}^{(gen)} \in S/\tilde{\sim} : S^{(gen)} \cap \tilde{S}^{(gen)} \neq \emptyset$ **do**
  $S/\tilde{\sim} \leftarrow (S/\tilde{\sim} \backslash \{S^{(gen)}, \tilde{S}^{(gen)}\}) \cup \{S^{(gen)} \cup \tilde{S}^{(gen)}\}$
end while

for $S^{(gen)} \in S/\tilde{\sim}$ **do**
  for $s \in S^{(gen)}$ **do**
    $[s]^{\tilde{\sim}} \leftarrow S^{(gen)}$ // map s to its equivalence class
  end for
end for

return $\tilde{\sim}, \langle S/\tilde{\sim}, S^{(ub)}/\tilde{\sim}, R/\tilde{\sim} \rangle$

Algorithm 5: GeneralizeModel

Data: $M = \langle S, S^{(ub)} \subset S, R \rangle : \sharp S = n, \sharp S^{(ub)} = \check{n} \leq n, \sharp R = m$ - metabolic model describing n metabolites, $\check{n}$ among them being ubiquitous, and m reactions.

Result: $\sim$ - approximation of the equivalence operation described in Problem 1, $M/\sim = \langle S/\sim, S^{(ub)}/\sim \subset S/\sim, R/\sim \rangle$ - corresponding generalized model.

$\mathring{\sim}, M/\mathring{\sim} \leftarrow Compute\mathring{\sim}(M)$

$\check{\sim}, M/\check{\sim} \leftarrow PreserveStoichiometry(\mathring{\sim}, M/\mathring{\sim})$

$\sim, M/\sim \leftarrow Maximize(\check{\sim}, M/\check{\sim})$

return $\sim, M/\sim = \langle S/\sim, S^{(ub)}/\sim, R/\sim \rangle$

Currently the generalization method depends on the ChEBI ontology. It cannot generalize metabolites that lack ChEBI annotations. In future work we will overcome this limitation.

The method zooms out a model to the most general level of abstraction that is consisted with the model structure, i.e., does not violate the restrictions 3.2 and 3.3. It remains to be seen whether there are intermediate levels of abstraction that can be useful for model analysis. In particular it may be interesting to define the maximal generalization for a group of organisms, in order to highlight the specific differences of the individual models with respect to a common generalization.

Appendix table 6.1, discussed in the next chapter (page 47), shows the results of the application of the model generalization method to 269 metabolic models from Path2Model project [Büchel et al., 2013].

The generalization method described in this chapter works well on metabolic networks that contain certain kinds of self-similarity, repeated patterns of reactions that operate on similar substrates and products with the same stoichiometry. While it is specifically designed for metabolic networks, the generalization algorithm does not depend on the metabolic origin of the network beyond the need for an ontology that labels its nodes. It could be potentially applied to any factorizable graph with an equivalent node labeling.

Given a bipartite graph whose nodes of one type are labeled by a trellis, generalization relabels sets of nodes with their least upper bounds, in a way that nodes of the second type with equivalently-labeled neighboring nodes can be factored. The factored (or compressed) graph contains one node per pattern of neighboring labels. Generalization preserves the in- and out-degrees of the nodes of the second type, and minimizes the degrees of the nodes of the first type in the compressed graph. While the result is not

```
<listOfSpecies>
  <species metaid= "m_s_0045" id="s_0045"
           name="(S)-3-hydroxydecanoyl-CoA [perox...]"
           compartment="c_14" ...>
    <annotation>
      <rdf:RDF xmlns:rdf="http://www.w3.org/1999/...">
        <rdf:Description rdf:about="#m_s_0045">
          <bqbiol:is><rdf:Bag>
            <rdf:li rdf:resource="ht.../chebi:28325"/>
          </rdf:Bag></bqbiol:is>
          ...
      </rdf:RDF>
    </annotation>
  </species>
  ...
<groups:listOfGroups>
  <groups:group metaid="m_s_gen_9" groups:id="s_gen_9"
                groups:name="hydroxy FA-CoA [pero...]"
                groups:kind="classification">
    <annotation>
      <rdf:RDF xmlns:rdf="http://www.w3.org/1999/...">
        <rdf:Description rdf:about="#m_s_gen_9">
          <bqbiol:is><rdf:Bag>
            <rdf:li rdf:resource="ht.../chebi:61902"/>
          </rdf:Bag></bqbiol:is>
          ...
      </rdf:RDF>
    </annotation>
    <groups:listOfMembers>
      <groups:member groups:idRef="s_0045"/>
      <groups:member groups:idRef="s_0051"/>
      <groups:member groups:idRef="s_0057"/>
      <groups:member groups:idRef="s_0054"/>
      <groups:member groups:idRef="s_0048"/>
      <groups:member groups:idRef="s_0236"/>
    </groups:listOfMembers>
  </groups:group>
  ...
```

Fig. 3.4 **Representation of a generalized model in SBML format with groups extension.** The output SBML file contains the initial model (including the lists of metabolites (called *species* in SBML), reactions, etc.) plus the *listOfGroups* section that represents non-trivial quotient metabolite and reaction sets. In the figure, a group representing a quotient metabolite set of *hydroxy fatty acyl-CoAs* is shown; it includes *(S)-3-hydroxydecanoyl-CoA* (s_0045), *(S)-3-hydroxylauroyl-CoA* (s_0051), etc. Each of those metabolites was previously declared in the *listOfSpecies* section.

guaranteed to be optimal, the compressed graph has been seen to be a good approximation of a graph with the minimal number of nodes of the second type, for the graphs that we analyze. Analysis of the 269 examples next in chapter 4 gives some idea of what

repetition, and consequently what generalization, is possible in real networks in nature.

An interesting open question for future work, related to the question of whether this kind of factoring is pertinent for other kinds of graphs, is what properties of the initial graph lead to this being a good approximation? Can we predict the degree of factorization from measures of subgraph similarity, or from properties of the trellis used for labeling? We have seen in practice that the constraint of preserving stoichiometry is essential for preserving the semantics of the network; more generally, one could ask what classes of topological constraints lead to better or worse rates of compression.

Since biological networks are often formed by specialization of existing networks, one could expect similar properties of generalization, and an immediate goal would be to test the generalization algorithm on cellular signaling networks and on transcriptional regulation networks.

Chapter 4

Validation of knowledge-based generalization

4.1 Applications

In order to demonstrate how the generalization method helps to detect problems and particularities in metabolic networks, we applied it to 1 286 metabolic networks that describe the same process in as many different organisms [Zhukova and Sherman, 2014b]. For our evaluation we chose *fatty acid metabolism*, both because it is a well-studied target for biotechnology applications, and because its presence or absence in different phylogenetic clades is generally known. We downloaded the networks that describe fatty acid metabolism from *Path2Models* [Büchel et al., 2013] project. *Path2Models* is a branch of the *Biomodels* database, that stores networks that were automatically generated from *KEGG* pathways.

The process of *β-oxidation of fatty acids* [Metzler and Metzler, 2001] repeats four main steps:

1. *dehydration*, transforming *fatty acyl-CoA* into *dehydroacyl-CoA*,
2. *hydration*, transforming *dehydroacyl-CoA* into *hydroxyacyl-CoA*,
3. *oxidation*, transforming *hydroxyacyl-CoA* into *3-oxoacyl-CoA*, and
4. *thiolysis*, transforming *3-oxoacyl-CoA* into *acetyl-CoA* and *fatty acyl-CoA* with a two carbons shortener chain.

A long chain of reactions, repeating these four steps again and again while transforming a long-chain *fatty acyl-CoA* into a short-chain one, becomes a cycle in a generalized

network: the reactions operating with the *fatty acyl-CoA* metabolites of different carbon chain length, corresponding to each of the steps, are factored together into four generalized reactions (see Figure 4.1).

Among the 1 286 networks that we have generalized, 243 do not have the *β-oxidation* pathway at all, and 124 have the complete *β-oxidation* cycle present.

4.1.1 Missing steps

If an enzyme catalyzing some of the reactions is missing in the network, then the generalized representation is not a cycle any more. For example, if *EC 1.1.1.35* is missing, the whole group of *oxidation* reactions participating in the network is eliminated, breaking the cycle (see Figure 4.2). This is more evident on a generalized network than on the initial one, where the absence of these reactions might be hidden by the abundance of other reactions.

Among the generalized networks, 128 have one step missing, 95 of them miss *oxidation*, 23 lack *dehydration*, 8 do not have *hydration*, and only 2 (*BMID000000046743* and *BMID000000129004*) miss *thiolysis*. As the most of the *β-oxidation* pathway is present, it is probable that the absence of this step is an error in the reconstruction process. For example, in network *BMID000000136479*, which represents *fatty acid metabolism* in the yeast *Yarrowia lipolytica (strain CLIB 122/E 150)*, the *oxidation* step is missing (Figure 4.2); while in the generalized network of the curated genome-scale network of the same strain of the same organism *MODEL1111190000* [Loira et al., 2012], the *β-oxidation* cycle is complete (Figure 4.3). By helping to draw the curator's attention to such missing steps, generalization can improve the speed and accuracy of network curation. Generalization can highlight missing steps by showing broken cycles, but also by showing changes in the *path profile*, the number of grouped reactions along a path (Figure 4.2).

On the other hand, 145 networks have two steps missing, and 646 have only one out of the four generalized reactions present.

4.1.2 Alternative steps

In addition to missing steps, the generalization of the network can highlight alternative paths that may be shortcuts or represent substrate specificities. It is important that such paths not be hidden in the generalized network, as they are often the cases that require the human expert's decision as to whether these alternatives appear due to an error or to an organism-specific adaptation.

In the case of *β-oxidation*, an example of reaction variations are two versions of the

oxidation reaction that use different ubiquitous metabolites, as shown in Figure 4.4. Among the networks that we analyzed, it is the only reaction that may have variations within the same network, indeed, 168 out of 170 networks that have the *oxidation* reaction present, have it in two versions.

Complete statistics on missing and alternative *β-oxidation* steps in the analysed networks are shown in Table 4.1. Changes in the numbers of grouped reactions in a profile path can also be used to evaluate alternative paths (data not shown).

4.2 Comparison of generalized networks

By abstracting detailed networks, generalization makes it easier to compare them at different scales of divergence. Since each generalization is maximal, as determined by the actual reactions and metabolites in the network, it masks unimportant differences in the intermediate levels of the ontologies of chemical entities and reactions. Stoichiometry preserving and metabolite diversity restrictions guarantee that any differences between two networks that remain after generalization result from real differences in their network structure. Furthermore, generalization makes these differences stand out from the structure of the conserved generalized network.

For example, the comparison of the standard *β-oxidation* pathway (Figure 4.1) and those for *Y. lipolytica* (Figure 4.3) and *B. thailandensis* (Figure 4.4) very clearly shows the specificities of the two latter networks, as well as the metabolites that prevent generalization. In *Y. lipolytica* C24:0-CoA is specially handled by specific acyl-transferase and fatty acid oxidation enzymes; in *B. thailandensis* a specific dehydrogenase is used for oxidation in one case out of six.

To explore the effect of network generalization on a broad evolutionary scale, we first mapped the 1 286 networks to the NCBI taxonomy database [Sayers et al., 2009] and compared *β-oxidation* pathway configurations between superkingdoms (Tables 4.1 and 4.2). The analysed networks represent *fatty acid metabolism* in 138 *eukaryota*, 1 045 *bacteria* and 103 *archaea* species.

The percentage of species for which the four-step *β-oxidation of fatty acids* is not present, or only one out of the four reactions is available (thus most probably used in a different pathway) is similar (about 60%) for all the superkingdoms. The case when the complete cycle is present diverges more. The complete cycle appears in some *eukaryota* and *bacteria*, but not in any of the 103 analysed *archaea* networks. This situation is supported by the MetaCyc pathway database: The *β-oxidation* pathway is present for *eukaryota* and *bacteria*, but not for *archaea*. This might be explained by the fact that in

spite of the presence in most *archaea* of the gene candidates for degradation of activated *fatty acids* via the *β-oxidation* pathway, *archaea* do not encode components of a *fatty acid synthase* complex [Falb et al., 2008].

Table 4.1 Presence of reactions of the *generalized β-oxidation of fatty acids* cycle in different networks **across the three superkingdoms** (•• stands for two versions of the corresponding reaction present in the network).

dehyd-ration	*hyd-ration*	*oxi-dation*	*thio-lysis*	number of all networks	eukaryota	bacteria	archaea
•	•	••	•	124	4	120	
•	•	-	•	95	33	44	18
-	•	••	•	23		23	
•	-	••	•	8		8	
•	•	••	-	2	1	1	
•	-	-	•	68	14	48	6
-	•	-	•	63		44	19
-	-	••	•	10		10	
•	•	-	-	2	2		
-	•	••	-	1	1		
•	-	•	-	1		1	
•	-	-	-	430	65	365	
-	-	-	•	166	13	93	60
-	•	-	-	49		49	
-	-	•	-	1	1		
-	-	-	-	243	4	239	
			Total:	1286	138	1045	103

Table 4.2 Percentage of different generalized *β-oxidation of fatty acids* cycle configurations in different networks.

β-oxidation cycle configuration	% of all networks	eukaryota	bacteria	archaea
complete cycle	10%	3%	11%	0%
one step missing	10%	25%	7%	18%
two steps missing	11%	12%	10%	24%
three steps missing	50%	57%	49%	58%
all steps missing	19%	3%	23%	0%

To further explore how generalization can help compare networks across evolutionary ranges, we considered the *β-oxidation* pathway in 47 fungal species (Table 4.3). The

first striking result is that the KEGG pathway method used by *Path2Model* seems to systematically miss the oxidation enzyme (column 3), since it is absent for almost all fungal networks yet fatty acid metabolism is a very common pathway. The second is that dehydration and thiolysis enzymes (columns 1 and 4) are almost always present, which is surprising, but since these are large classes of enzymes that are present in other pathways, perhaps many of the enzymes in these columns are misassigned to this pathway. What remain are the hydration enzymes (column 2), which show some variation between the networks in Table 4.3. In many cases these enzymes are absent in known pathogens, such as the *Candida*, which hints that these species may obtain the fatty acids from the host rather than through synthesis. However, the systematic biases seen in the other columns make it impossible to find the correlated gene losses that are the hallmark of missing pathways.

Significantly, this shows that network generalization is an excellent tool for abstracting networks from very different lifestyles up to a comparable level of complexity, that directly reveals species-specific differences and systematic biases. These are precisely the clues that human curators would need in order to judge to what degree the *β-oxidation* pathway is present in each of these species.

4.3 Detection of generalization profile classes

To illustrate how the model generalization performance on the genome-scale networks, we applied it to other 269 metabolic models from Path2Model project [Büchel et al., 2013]. All those models are genome-scale, and the the average number of reactions per model is 2 879.

Appendix Table 6.1 shows the numbers of reactions in those models before and after the generalization. The average compression ratio ($\frac{\#\ reactions\ in\ initial\ model}{\#\ reactions\ in\ generalized\ model}$) is 1.14, but as we will see this is misleading because the distribution is heavily skewed.

Call a *generalization profile* of a model an integer vector that at each index i contains the *generalization ratio*, the number of reactions in the initial model that formed a group of exactly i similar reactions during the generalization. For example, for a model containing 15 reactions, 6 of which formed 3 pairs of similar reactions, 4 formed a group of 4 similar reactions, and 5 reactions were not generalized, the generalization profile is represented by a vector $[5,6,0,4]$. In the case of 269 genome-scale models that we used for our analysis, the similar reaction group of the largest size was found in the model BMID000000140362 (the model of the whole-genome metabolism of the bacterium *Rhodococcus sp.* RHA1) and contained 40 reactions. Thus, the generalization profile vectors

were of length 40. The *generalization matrix* is the matrix whose rows are models and columns are generalization ratios.

A generalization profile is essentially a histogram of a discrete distribution of generalization ratios. A typical such distribution is strongly positively skewed and leptokurtic: most generalizations concern only 1–3 reactions, but there is a long right tail of a relatively small number of generalized reactions that concern a large number of reactions (up to 40 in the example above). The large number of slightly generalized reactions is a common feature of the profiles. Let us investigate whether the long right tails are uniformly shaped, or whether they permit to divide the generalization profiles into classes.

In order to avoid bias from the leftmost positions in the profile whose frequencies are consistently high, we first scale the profiles. For each column in the generalization matrix, we first center the data by subtracting the column mean from the values, then we scale the column by dividing each value by the standard deviation of the column. The distance between two profiles in the generalization matrix is the Euclidean distance between the two vectors defined by the centered, scaled data in the two corresponding rows.

To investigate whether there exists a collection of different shapes for the long right tails, we computed *self-organizing maps* [Kohonen, 1982] (SOMs) of the profiles using the som method of the R package kohonen. A self-organizing map is a non-linear partitioning method that creates a map in which similar observations are grouped, and groups with similar patterns are positioned next to each other in the map. The particularity of an SOM is that, during training, observations are moved to neighboring groups. It produces a low-dimensional (typically two-dimensional), discretized representation of the input space of the training samples, and is therefore useful for visualizing low-dimensional views of high-dimensional data. Figure 4.5 shows the resulting SOMs on a 8×6 and 2×2 grids. The first shows that different patterns of generalization profile exist: some have little generalization, but many profiles are complex. The second is an exaggerated simplified view that shows only four classes: two with some or little generalization, two with significant generalization of different kinds.

While this shows that different classes of profile exist, much further investigation will be necessary to understand the link between the similarity of two generalization profiles, and the similarity between the corresponding two models. Since generalization often creates a cycle from metabolic pathways built by successive, similar chains of reactions, a straightforward approach to start with would be to search for similarities in the pathways of the organisms in the same SOM class.

4.4 Discussion

Using our network generalization method we have studied 1 286 networks describing metabolism of fatty acids in as many organisms. Generalization helps a human to understand, compare and classify those networks. Providing a higher-level view of the network by factoring the abundance of similar reactions, it allows for easier comprehension of the general network structure, and highlights possible problems and organism-specific particularities. Generalization highlights potential errors in inferred draft networks, exposes specific absences or alternatives, and makes it possible to compare networks between species, clades, and kingdoms at a higher level of abstraction.

Generalization can also help in finding a standard template for a pathway, using which curators can analyze this pathway in the organism of interest. In our example, the complete *β-oxidation* cycle without alternative steps, served as a standard template.

We have studied the correlation between differences in the generalized networks, and their belonging to one of the three superkingdoms: *eukaryota*, *bacteria* or *archaea*. Our method highlighted known tendencies of these superkingdoms, such as the absence of *β-oxidation* in *archea*. However, a far more interesting goal is to understand the differences between networks of closely related organisms, in order to study the connection between the differences in generalized metabolic properties of organisms and the differences in their physiology in more refined details, comparing phylogenetically close species, or even different strains of the same organism. Network generalization will expose the absent reactions or the alternative paths that characterize individual species or strains, and more closely establish the link between genotype and phenotype.

We have computed the generalization profiles of 269 genome-scale metabolic models and applied the self-organizing map method to classify them. We detected several distinct generalization profiles. An interesting future work would be to investigate the correlation between those classes and the biological characteristics of the networks.

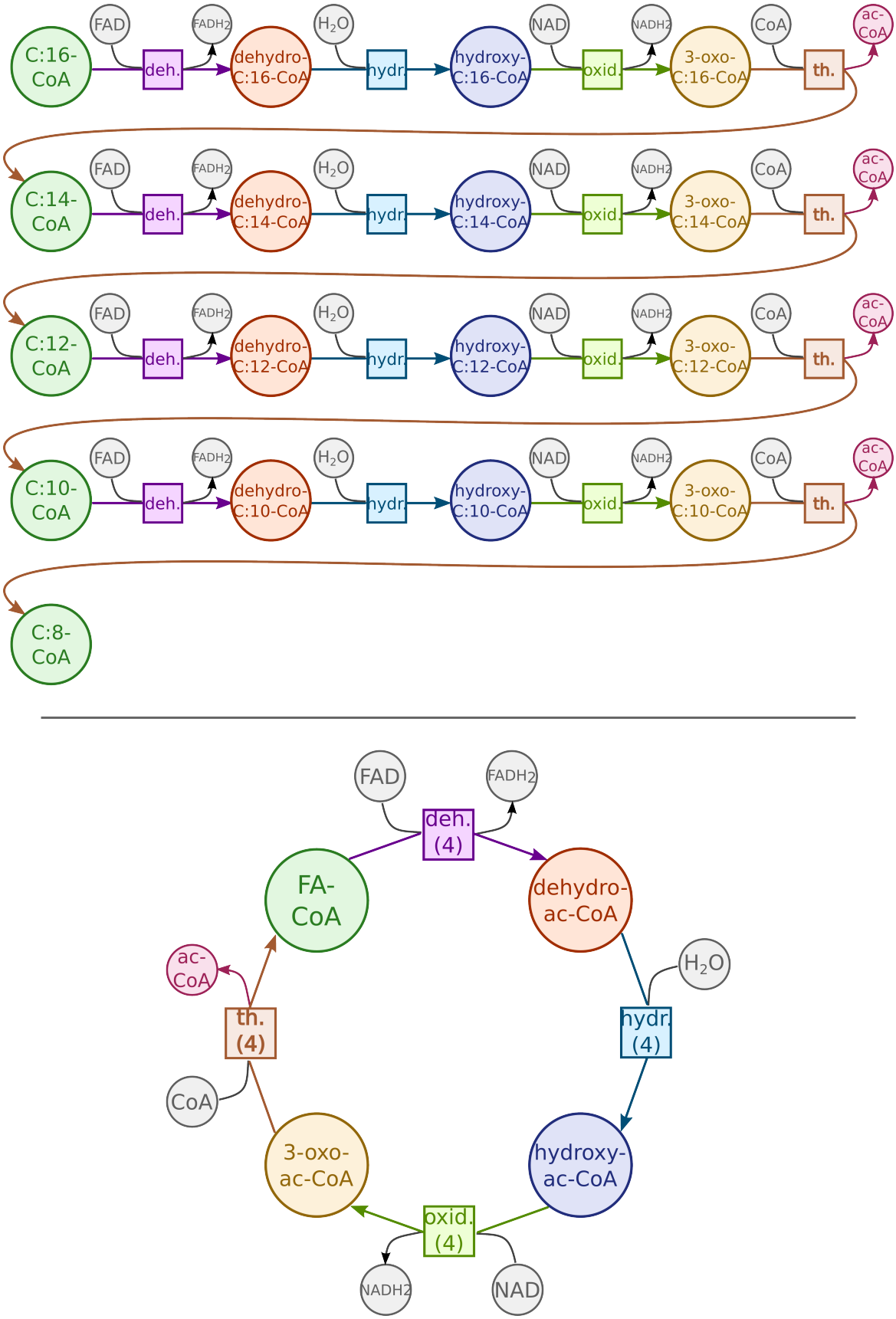

Fig. 4.1 **Generalization of *β-oxidation of fatty acids*.** The initial representation of the of *β-oxidation of fatty acids* pathway (top) and its generalized representation (bottom). The number in parentheses in each generalized reaction shows how many specific reactions were grouped together.

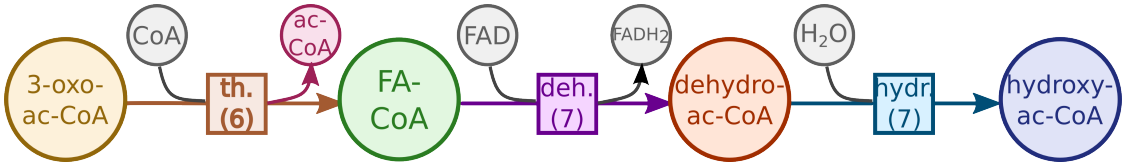

Fig. 4.2 **Missing reactions.** The generalized representation of *β-oxidation of fatty acids* of *BMID000000136479* (oleaginous yeast *Y. lipolytica*, noncurated network from *Path2Models*). The *oxidation* reaction is missing.

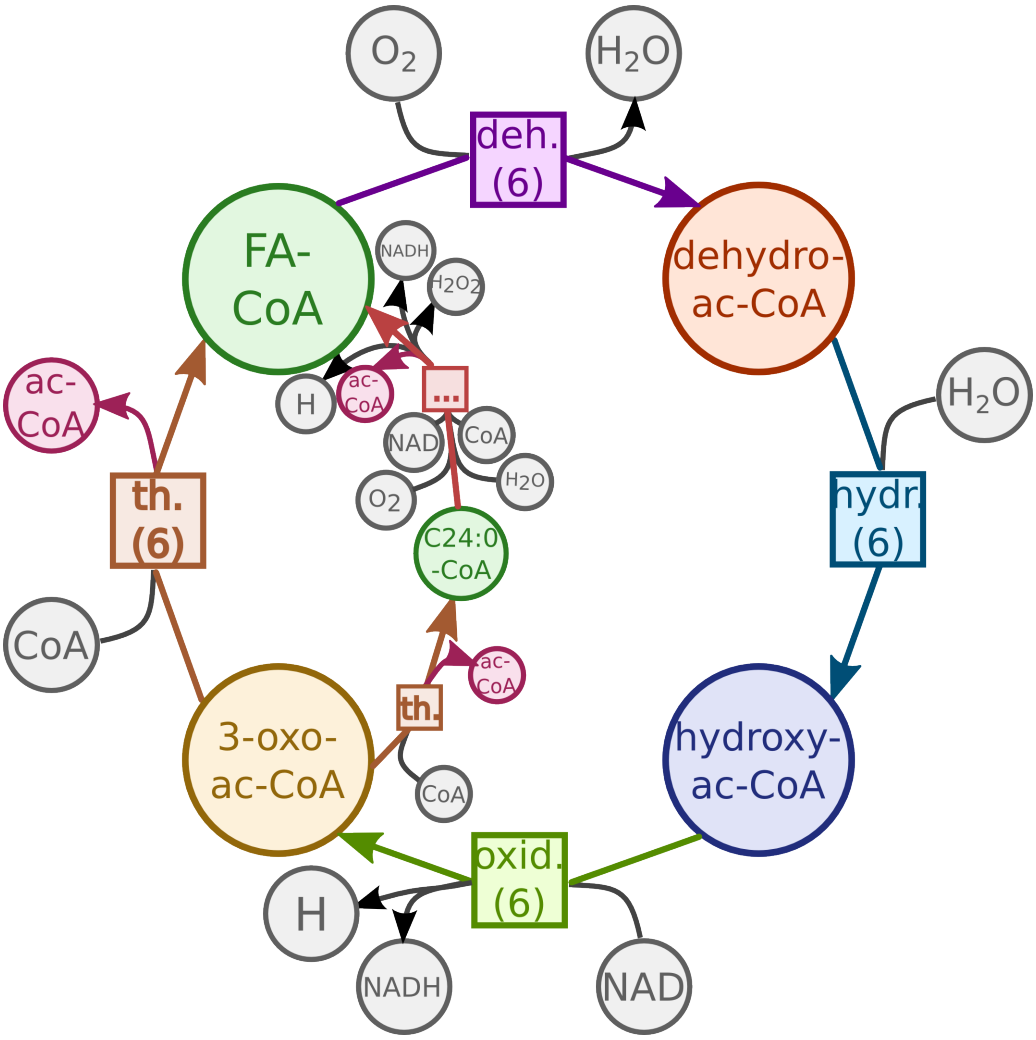

Fig. 4.3 **Generalization of *β-oxidation of fatty acids* of *MODEL1111190000*** (*Y. lipolytica*, curated network from [Loira et al., 2012]). The cycle is complete.

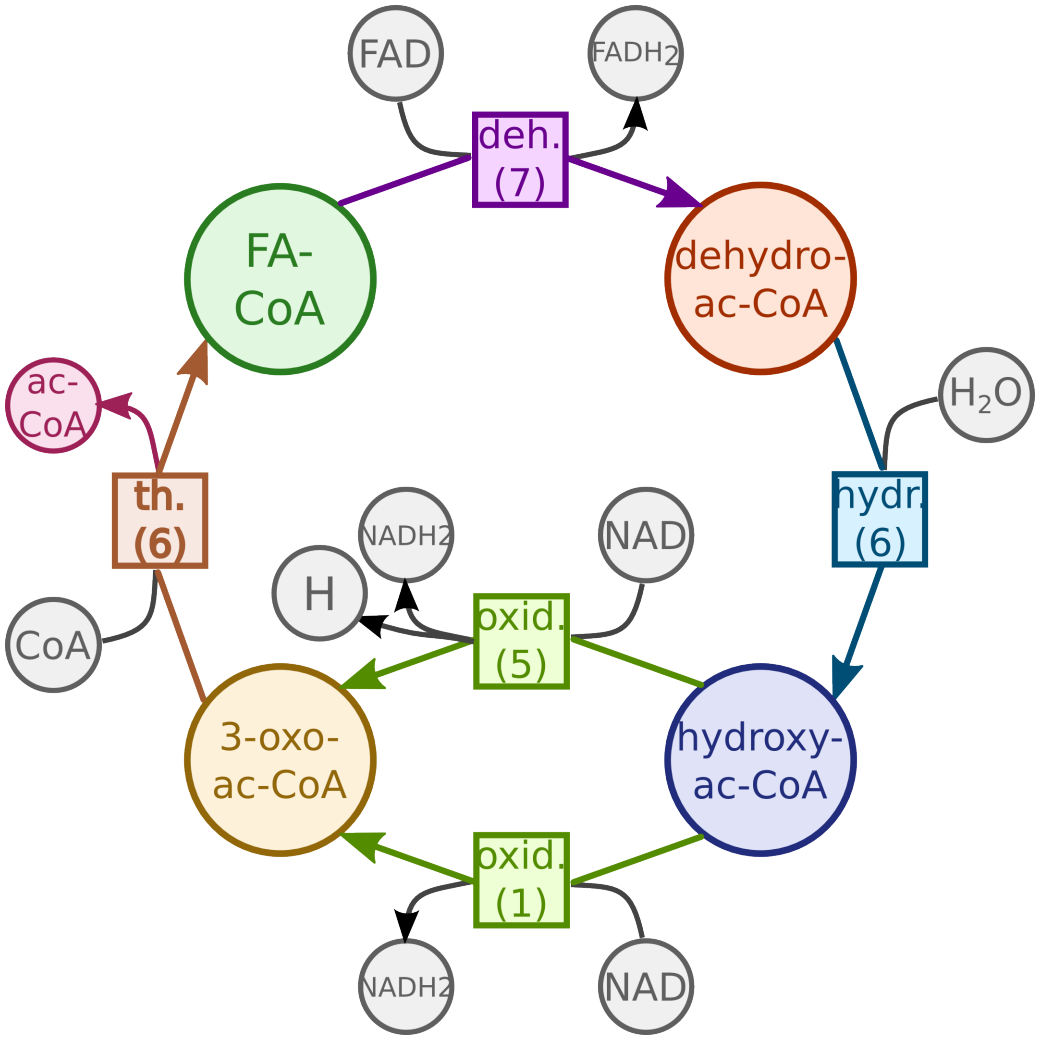

Fig. 4.4 **Alternative paths.** The generalized representation of *β-oxidation of fatty acids* of *BMID000000103487* (nonpathogenic bacterium *Burkholderia thailandensis*). Two variants of the *oxidation* reaction (bottom) are present.

Table 4.3 Presence of reactions of the *generalized β-oxidation of fatty acids* cycle in different networks of **fungal genomes**.

fungal species	*dehydration*	*hydration*	*oxidation*	*thiolysis*
Microsporidia				
. Encephalitozoon cuniculi GB-M1	-	-	-	•
Dikarya/Ascomycota				
. Taphrinomycotina				
. . Schizosaccharomyces pombe 972h-	-	-	-	•
. saccharomyceta				
. . Saccharomycotina/Saccharomycetales				
. . . Metschnikowiaceae				
. . . . Clavispora lusitaniae ATCC 42720	•	-	-	•
. . . **Debaryomycetaceae**				
. . . . Lodderomyces elongisporus NRRL YB-4239	•	-	-	•
. . . . Scheffersomyces stipitis CBS 6054	•	•	-	•
. . . . Meyerozyma guilliermondii ATCC 6260	•	-	-	•
. . . . Debaryomyces hansenii CBS767	•	•	-	•
. . . **Dipodascaceae**				
. . . . Yarrowia lipolytica CLIB122	•	•	-	•
. . . **Saccharomycetaceae**				
. . . . Komagataella pastoris GS115	•	-	-	•
. . . . Zygosaccharomyces rouxii CBS 732	•	-	-	•
. . . . Lachancea thermotolerans CBS 6340	•	-	-	•
. . . . Saccharomyces ceremonial S288c	•	-	-	•
. . . . Vanderwaltozyma polyspora DSM 70294	•	-	-	•
. . . . Ashbya gossypii ATCC 10895	•	-	-	•
. . . . Candida glabrata CBS 138	•	•	-	•
. . . . Kluyveromyces lactis NRRL Y-1140	•	•	-	•
. . . **mitosporic Saccharomycetales/Candida**				
. . . . Candida dubliniensis CD36	•	-	-	•
. . . . Candida tropicalis MYA-3404	•	-	-	•
. . . . Candida albicans SC5314	•	-	-	•
. . Pezizomycotina				
. . . Pezizomycetes				
. . . . Tuber melanosporum Mel28	•	•	-	•
. . . leotiomyceta				
. . . . dothideomyceta/Phaeosphaeria nodorum SN15	•	•	-	•
. . . . **sordariomyceta**				
. . . . . Leotiomycetes/Sclerotiniaceae				
. . . . . . Sclerotinia sclerotiorum 1980 UF-70	•	•	-	•
. . . . . . Botryotinia fuckeliana B05.10	•	•	-	•

Table 4.3 (*Continued*). Presence of reactions of the *generalized β-oxidation of fatty acids* cycle in different networks of **fungal genomes**.

fungal species	*dehyd-ration*	*hyd-ration*	*oxi-dation*	*thio-lysis*
. . . . . Sordariomycetes				
. . . . . Sordariomycetidae				
. . . . . Podospora anserina S mat+	•	•	-	•
. . . . . Neurospora crassa OR74A	•	•	-	•
. . . . . Magnaporthe oryzae 70-15	•	•	-	•
. . . . . Hypocreomycetidae				
. . . . . Fusarium graminearum PH-1	•	-	-	-
. . . . Eurotiomycetes/Eurotiomycetidae				
. . . . . **Eurotiales/Aspergillaceae**				
. . . . . Penicillium chrysogenum Wisconsin 54-1255	•	-	-	-
. . . . . Neosartorya fischeri NRRL 181	•	•	-	•
. . . . . Aspergillus oryzae RIB40	•	-	-	-
. . . . . Aspergillus niger CBS 513.88	•	-	-	-
. . . . . Aspergillus clavatus NRRL 1	•	•	-	•
. . . . . Aspergillus flavus NRRL3357	•	-	-	-
. . . . . Aspergillus fumigatus Af293	•	•	-	•
. . . . . Aspergillus nidulans FGSC A4	•	-	-	-
. . . . . **Onygenales**				
. . . . . Uncinocarpus reesii 1704	•	•	-	•
. . . . . Coccidioides immitis RS	•	•	-	•
. . . . . Coccidioides posadasii C735 delta SOWgp	•	•	-	•
Dikarya/Basidiomycota				
. **Ustilaginomycotina**				
. . Malassezia globosa CBS 7966	•	-	-	-
. . Ustilago maydis 521	•	-	-	-
. Agaricomycotina				
. . **Agaricomycetes**				
. . . Postia placenta Mad-698-R	•	•	-	-
. . . Schizophyllum commune H4-8	•	•	-	•
. . . Moniliophthora perniciosa FA553	•	•	-	•
. . . Laccaria bicolor S238N-H82	•	•	-	•
. . . Coprinopsis cinerea okayama7♯130	•	•	-	•
. . **Tremellomycetes/Cryptococcus neoformans**				
. . . Cryptococcus neoformans var. neoformans B-3501A	•	-	-	-
. . . Cryptococcus neoformans var. neoformans JEC21	•	-	-	-

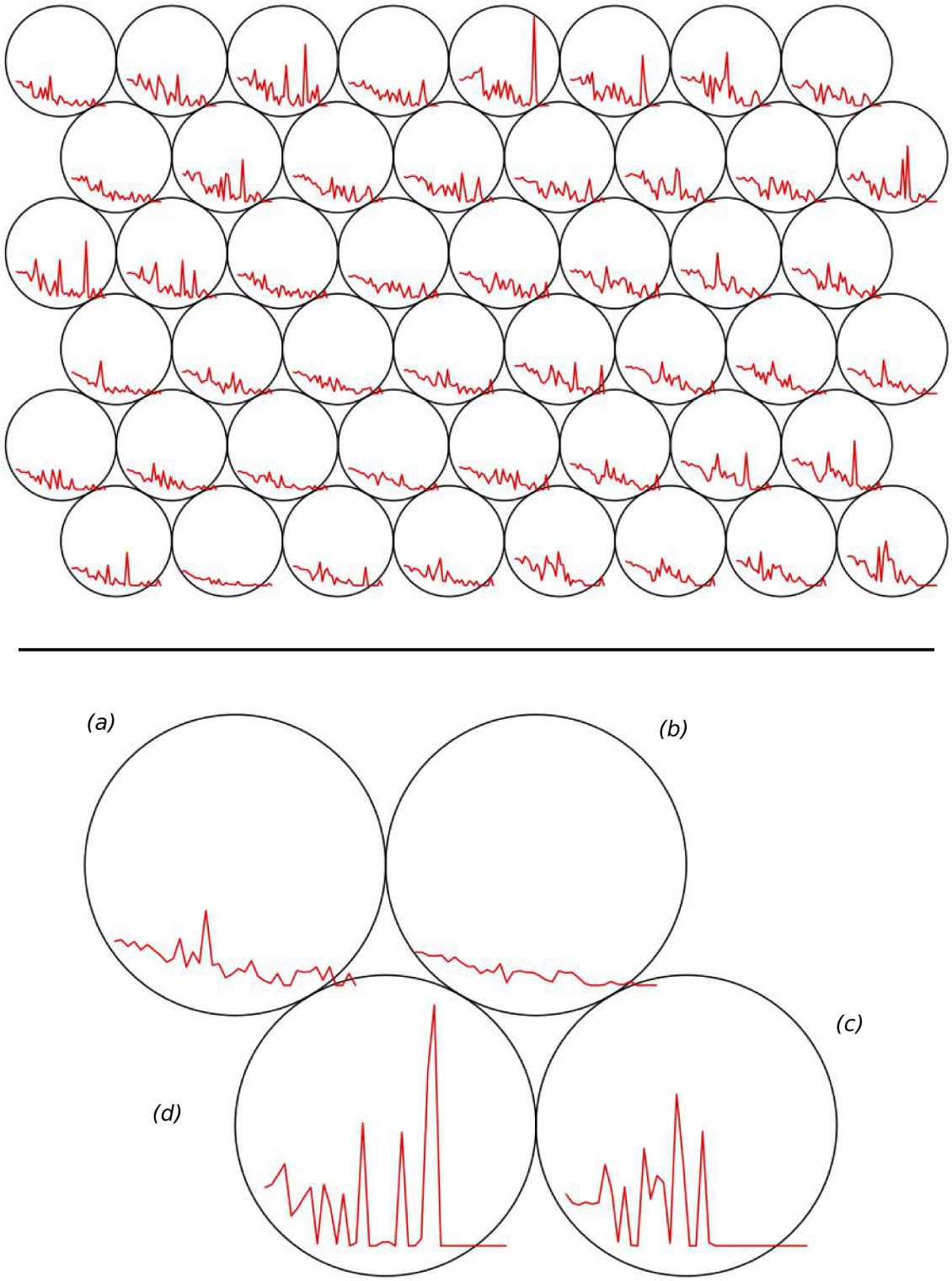

Fig. 4.5 **The self-organizing maps (SOMs) of model generalization profiles**. The 8×6 SOM (top) shows that there exist distinct classes of profile forms. For example, it shows that the right tail, after scaling, has a lot of influence on the classification. The 2×2 SOM (bottom) detects the 4 main classes of profile forms: *(a)* some generalization around 10-15; *(b)* almost no generalization; *(c)* significant generalization peaking at 10-25; *(d)* significant generalization with an additional peak at 25-35.

Chapter 5

Mimoza: web-based semantic zooming and navigation in metabolic networks

5.1 Background

In chapter 3 we have defined a theoretical model generalization method designed to aid users in understanding complex metabolic networks. Generalization identifies and groups similar metabolites and similar reactions in the network. Applied to different models, it can bring them to the same level of abstraction so that they can be compared, as we have shown in Chapter 4. To further explore the opportunities of the method we implemented it as a practical tool [Zhukova and Sherman, 2015].

The *zooming user interface* (ZUI) [Bederson and Meyer, 1998] paradigm has proven to be a powerful tool for representation of data at different scales. It is being adopted for various domains of applications, including cartographic [Nivala et al., 2008], exploratory data visualization [Roberts, 2005], collaborative interfaces [Laufer et al., 2011], and biological data [Hu et al., 2007]. The challenge is how to use ZUI-based visualization for semantic generalization of metabolic models.

5.1.1 Existing visualization approaches

In Chapter 2 we described various tools for model visualization. They include desktop tools (e.g., CellDesigner [Funahashi et al., 2008], VANTED [Rohn et al., 2012], Cytoscape [Smoot et al., 2011]) and web-based tools (e.g., JWS online [Snoep and Olivier, 2003], MetDraw [Jensen and Papin, 2014]) that produce reasonably good visualizations of small networks (up to hundreds of reactions), but become cluttered at the genome-scale level, making the visualization unreadable. Due to the huge numbers of metabolites and reac-

tions in genome-scale metabolic networks, we have an uncomfortable choice between either many edge crossings in an automatic visualization, or over-duplication of various metabolites making the essential parts of the model disconnected and the visualization too large to grasp.

We concluded that a different visualization approach is needed and proposed the *Zooming User Interfaces (ZUIs)*, which can change the nature of the content displayed at different zoom levels, as a pertinent alternative. ZUI can provide two main types of magnification: *geometric zooming*, in which a region of the network is enlarged; and *semantic zooming*, in which additional properties are introduced with enlargement [Hu et al., 2007].

We discussed in Chapter 2 several ZUI tools for visualization of biological data, including several ZUI tools that permit the visualization of metabolic networks: the Genome Projector [Arakawa et al., 2009], NaviCell [Kuperstein et al., 2013], the Cellular Overview [Latendresse and Karp, 2011] and the Reactome pathway database [Croft, 2013; Milacic et al., 2012] browser. Table 5.1 summarizes the main characteristics of these visualization tools.

None of those ZUI tools, except for NaviCell, allow users to input their own models. Moreover, as their examples show, not only geometric zoom but also model decomposition and semantic zoom are important for multi-level visualization of huge models. At the general level, the network needs to be decomposed into several meaningful modules (such as compartments, pathways). If after such a decomposition the model remains complicated, a further decomposition is required. We address these issues below by combining model generalization with a ZUI.

5.2 Implementation

Choosing zoom levels

We address the problem of large-scale metabolic model visualization by combining meaningful decomposition into modules with automatic multi-level abstraction. Decomposition is performed in the following way: The network is first split into compartments; then the model generalization method is applied to each compartment to detect the generalized modules. Thereby, the most appropriate is to adopt 3 levels of semantic zooming:

1. The most abstract level represents compartmentalization of the network, and focuses on such questions as: Are all the compartments present? Are they well connected by transport reactions?

This level shows the compartments of the model, the transport reactions between them, and other reactions happening inside the cytoplasm. If the model does not describe compartments, this level will be missing.

2. The second level shows the modules inside each of the compartments. The questions that can be addressed at this level include: Are all the reactions or more generally pathways desired by the curators present? are the input-output relations of functional modules consistent with what the expert expects from her knowledge? Does the model show organism-specific adaptations, seen in the model as shortcuts or meanders?

   We use our knowledge-based generalization method to identify the modules inside the compartments. It detects similar metabolites and reactions and clusters them together to represent them as generalized metabolites and reactions with the same structure (numbers of consumed and produced metabolites). The generalized representation reveals the overall structure of the network while hiding the details.

   If no similar metabolites/reactions can be detected by the generalization method (due to the model structure or to missing ChEBI metabolite annotations), this level will be missing.

3. The most detailed level is intended for computer simulation and represents the inner structure of each of the modules with all the metabolites, reactions and their kinetics, stoichiometries and constraints.

   Our method places similar metabolites and reactions (detected at level 2) next to each other, thus simplifying the analysis of their presence.

Figure 5.1 shows such a 3-level representation on the example of the model of β-oxidation of fatty acids [Metzler and Metzler, 2001] in the peroxisome compartment of a yeast *Y. lipolytica*. The first level (bottom) shows the peroxisome compartment, and the transport reactions; the second level (middle) shows the generalized structure of the peroxisome, the main processes happening in it; the most detailed level (top) represents the complete model, placing semantically similar metabolites and reactions next to each other.

5.2.1 Layers Layout

To visualize a metabolic network we first represent it as a bipartite graph [Diestel, 2012] with two disjoint sets of nodes (metabolites and reactions), and edges that connect the

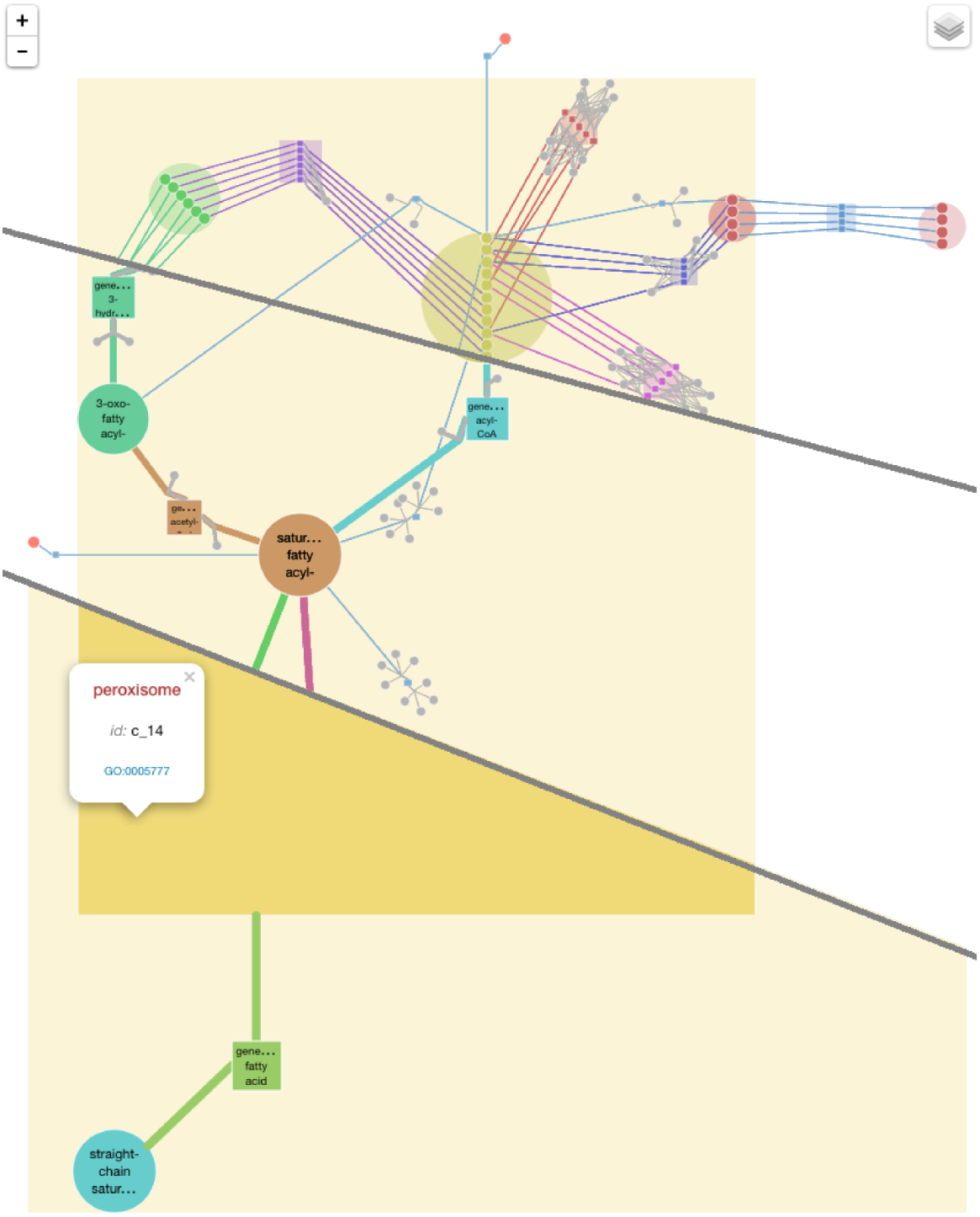

Fig. 5.1 **Three zoom levels** The most general zoom level (bottom) shows the peroxisome and a generalized transport reaction. The intermediate zoom (middle) shows the generalized processes inside the peroxisome compartment. The most detailed view (top) reveals the metabolites and reactions of the initial model.

reactions to their substrate and product metabolites. To achieve such a representation, we implemented a converter from SBML to TLP format, that is used by the Tulip graph visualization tool [Auber, 2004]. TLP format stores nodes and edges of the graph, and associates each node and edge to a list of named attributes: standard ones, such as shape, size, color; and user-defined ones, such as, in our case, element type (compartment, reaction or metabolite), ChEBI identifier, group number, gene association, etc. The SBML-to-TLP converter is implemented in python, using libSBML library [Bornstein et al., 2008], and is available as a part of Mimoza software.

While layout of large graphs is widely studied [Unwin et al., 2006], the correspondence between the layouts of different semantic zoom levels remains a hard task. To compute the layout for different semantic zoom levels we combine two different approaches.

5.2.1.1 Generalized model layout

In order to lay out the sub-networks corresponding to each of the compartments after the generalization, we use a combination of standard layout algorithms provided by Tulip. We divide the compartment graph into connected components (subgraphs in which any two nodes are connected to each other by undirected paths, and which are not connected to any additional nodes in the supergraph), using a method provides by Tulip. We then apply an appropriate layout algorithm on each of them. The results are combined together using the *Connected Component Packing* algorithm (provided by Tulip), which places the components close to each other while removing the overlaps between them.

Depending on the nature of the connected component subgraph, we choose one of the following layout algorithms, provided by Tulip:

- *Hierarchical Layout* for the components that contain no cycles (*Sugiyama (OGDF)* [Sugiyama et al., 1981] algorithm, that has the complexity of $O(|V||E|)$ in time and of $O(|V| + |E|)$ in space);
- *Circular Layout* for the components with less than 100 nodes and less than 3 cycles (*Circular (OGDF)* [Tamassia, 2007], with $O(|E|^2)$ time and space complexity);
- *Force-Directed Layout* for all the other components (FM^3 *(OGDF)* [Hachul and Jünger, 2005], that has the asymptotic worst-case running time of $O(|V|log|V| + |E|)$ with linear memory requirements).

To avoid clutter we duplicate all the *minor* metabolites (*oxygen*, *hydrogen*, *water*, *ATP*, etc.) before applying the layout algorithms, so that there is a copy of a minor metabolite for each reaction in which it is used. We then extract a subgraph, containing all but the

minor metabolites, apply the combined layout on it, and then place the minor metabolites next to the reactions in which they participate.

5.2.1.2 Generalization-based full model layout

The layout for the full model is based on the corresponding generalized model's layout. To allow zooming into the generalized model, we keep the same coordinates as in the generalized model for the minor metabolites and the ungeneralized metabolites and reactions, and place similar metabolites or reactions next to each other inside the space used by the corresponding generalized metabolites or reactions in the generalized model.

An edge in the generalized view might expand into several edges in the full-model view, for example, if it is a generalized edge connecting a generalized metabolite to a generalized reaction. The positions of the edges after such an expansion might slightly differ from the corresponding generalized one.

5.2.1.3 Node colors

A different color is assigned to each generalized metabolite/reaction; and is propagated to the corresponding metabolites/reactions of the full model. Minor metabolites are colored grey. Mimoza's interface includes a checkbox that permits to hide/show minor metabolites .

5.2.1.4 Node sizes

The size of the nodes depends on their nature: minor metabolites are smaller than the other ones; a radius of a generalized metabolite/reaction is calculated as a sum of radiuses of the elements that it groups; compartment sizes are defined by the layouts of the elements inside them, so that the compartments are represented as minimal rectangles containing all the corresponding elements. All major specific (i.e., not generalized) metabolites are of the same size; as well as all specific reactions.

5.2.1.5 Relative positions of compartments

Metabolic models may include several compartments, nested into each other. For example, the *peroxisome* compartment is surrounded by its *membrane*, and contained in *cytoplasm*; the *cytoplasm* is part of the *cell*, which is surrounded by the *cell envelope*.

SBML allows to represent relative positions of the compartments in the model with an optional *outside* tag. However, it is not available in all SBML levels, nor is widely used.

To be able to visualize the compartments correctly even for the SBML models lacking this information, we infer their relative positions from the Gene Ontology (GO) [Ashburner et al., 2000]. We associate each compartment with a term from the *cellular component* branch of GO by using annotations in the model if they are present, or matching the compartments' names otherwise. We then use the *part_of* and *is_a* relationships between the terms in GO to infer relative compartment positions. If no term for a compartment could be found, it is placed on the outer-most level.

5.2.1.6 SBML layout

To store the calculated layout of the model elements we use the layout extension [Gauges et al., 2013] of SBML. It allows to store the coordinates and sizes of the metabolites, reactions and compartments in the model. The TLP-to-SBML layout converter is implemented in python and is available as a part of Mimoza software. If the SBML model submitted by the user contains the layout information, our software uses it for nodes' positions. Therefore, it is possible to visualize a model with Mimoza, download the resulting SBML with layout annotations, edit it manually or with another software and then revisualize the updated version with Mimoza.

5.2.2 ZUI

The zoomable interactive representation is achieved using Leaflet [Agafonkin, 2010], a JavaScript library for interactive maps.

We export elements of the network graph (compartments, metabolites and reactions) as map features in GeoJSON format [Butler et al., 2008] in order to store their coordinates and metadata (e.g., ChEBI annotations for metabolites). Figure 5.2 shows an example of a reaction represented in GeoJSON format. The TLP-to-GeoJSON converter is implemented in python and is available as a part of Mimoza software.

The GeoJSON objects are then added as layers to the map and rendered by Leaflet into clickable elements at corresponding zoom levels. We follow SBGN Process Description language convention [Le Novère et al., 2009b] to choose the glyphs for model elements' representation: Metabolites are drawn as circles linked by edges to the reactions where they participate; reactions are represented as squares; compartments are drawn as rectangles. When a user clicks on a map element a pop-up appears (see Figure 5.3) showing its name, identifier and additional information, e.g. gene associations and formulas for reactions. Two overlays allow user to show or hide minor metabolites (e.g., water, oxygen, hydrogen, etc.), and transport reactions.

```
{
  "geometry": {
    "type": "Point",
    "coordinates":[363.53, 179.46]
  },
  "type": "Feature",
  "properties": {
    "c_id": "c_14",
    "term": [
      ["YALI0E18568g"]
    ],
    "name": "acetyl-CoA acyltransferase",
    "rs": [
      ["3-oxooctadecanoyl-CoA [peroxisome]", 1],
      ["coenzyme A [peroxisome]", 1]
    ],
    "color": "#79A8C9",
    "rev": true,
    "ps": [
      ["acetyl-CoA [peroxisome]", 1],
      ["palmitoyl-CoA [peroxisome]", 1]
    ],
    "w": 1.28,
    "type": 2,
    "id": "r_0115"
},
```


Fig. 5.2 **GeoJSON representation of a reaction.** An SBML reaction is stored as a GeoJSON Point feature, with its layout coordinates encoded in the geometry section. The identifiers, labels and annotations, as well as the information on the reactant and product metabolites are stored as properties. The "type" property value specifies that this GeoJSON feature is a reaction.

5.2.3 Embedding

After the visualization with Mimoza is done, we provide a link for embedding the view in another web page.

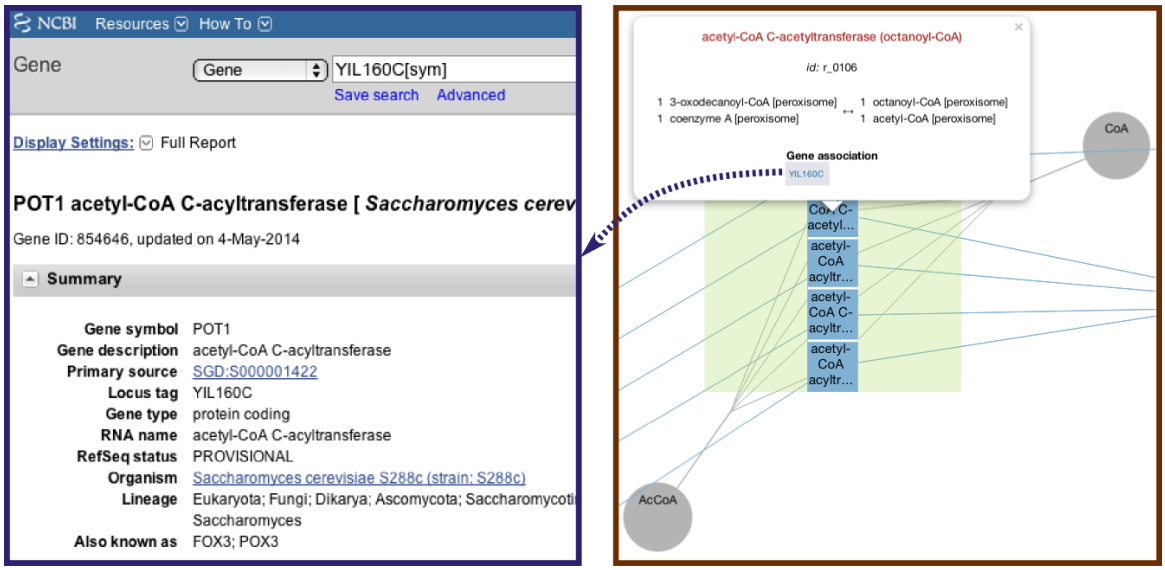

Fig. 5.3 **A reaction pop-up.** (right part) An example of a pop-up that opens when a user clicks on a reaction: It contains the information on the reaction name, identifier, reactant and product metabolites and their stoichiometries, as well as gene associations. (left part) Gene names are hyperlinks redirecting to the NCBI Gene database [NCBI, 2014].

5.2.4 Download and distribution

One can use Mimoza in three different ways:

1. As a standalone application. All Mimoza code is open-source and can be downloaded from the project web page [Zhukova and Sherman, 2014c] and installed on a local server.

2. On the Mimoza web server. Mimoza web server [Zhukova and Sherman, 2014c] lets one test visualization for smaller SBML models, with the possibility to download the result as a COMBINE archive [Bergmann et al., 2014], including the SBML file with groups (to store the metabolite and reaction groupings) and layout (to store the element coordinates) extensions, GeoJSON files with the coordinates of model elements, and the HTML, CSS and JavaScript files that are needed to view the visualization in a browser.

3. As a Galaxy [Blankenberg et al., 2010] project tool, so that generation of Mimoza views can be included in a Galaxy workflow. The Galaxy wrapper for Mimoza is available for download from the project web page.

5.2.5 Pipeline

The overall Mimoza pipeline contains 5 steps:

1. The user submits a model in SBML format (level 2 or 3, any version) via a web form.

2. If the model does not yet contain groups, it is generalized using the model generalization method, and the resulting SBML file (level 3 version 1 with groups extension) is made available to the user.

3. The SBML file with groups of similar metabolites and reactions is converted into a Tulip graph: metabolite nodes are connected by edges to the nodes of the reactions in which they participate. The generalized metabolites and reactions form quotient nodes. The Tulip graph is split into sub-graphs corresponding to different compartments, and layout algorithms are applied to them.

4. The compartment sub-graphs are exported in GeoJSON format and rendered by the Leaflet library into an interactive map that is represented to the user.

5. The result can either be browsed on the Mimoza web page directly, or downloaded as a COMBINE archive and embedded into a different website.

5.3 Results and Discussion

To illustrate the use of Mimoza and compare it with other available ZUI tools, we visualized the yeast consensus genome-scale metabolic network model [Herrgå rd et al., 2008]. The result can be found at http://mimoza.bordeaux.inria.fr/yeast4/comp.html?id=C_1. Mimoza automatically split the network into compartments and created a 3-level visualization for each of them.

We visualized the same model using MetDraw with no manual adjustments. The resulting SVG file (http://www.metdraw.com/metdraw/bc7df60221ba314c383b1bf6e7dad4c3056f92bb) has only one zoom level with lots of clutter, that does not allow one to see the structure of the network.

Cellular Overview does not allow one to visualize a model provided by a user, but has a map of metabolism of *Saccharomyces cerevisiae*: http://biocyc.org/overviewsWeb/celOv.shtml. It has a clear non-overlapping representation of various pathways present in the model, but does not show the compartmentalization. It is not automatic and is pathway-oriented, thus is not suitable for models having no pathway metadata. The

zoom-in shows additional labels but all the metabolites and reactions are present at all the levels, making the elements at the most general level very small and hard to analyze.

NaviCell does not allow to visualize an SBML model automatically. Genome Projector only contains maps for bacterial genomes and does not permit user's model input.

Neither Reactome allows users to visualize their own models, but it contains a pathway map for *Saccharomyces cerevisiae*: http://www.reactome.org/PathwayBrowser/#SPECIES=68322&DIAGRAM=5662370. It has two semantic zoom levels: a visualization of a list of pathways present in the model, and submaps corresponding to each of them. The representation of each pathways is very clear, and has several geometric zoom levels. However, it is not always space-efficient as it contains gaps due to reactions present in other organisms but absent in *S. cerevisiae*. Another particularity is that while the positions of elements common to different organisms are conserved within a pathway, their positions might differ between different pathways of the same organism. In Mimoza, on the contrary, the positions of the reactions and metabolites are conserved between the compartments of the same organisms; but the layout of common processes (e.g. pathways) in different organisms' visualizations might differ in the current implementation.

Table 5.1 summarizes the comparison of Mimoza to other ZUI tools. Mimoza especially targets draft models during curation, allowing one to visualize them fully automatically and helps to analyze them in a top-down manner, starting from the general structure and going down to the details. The generalized level differentiates it from other tools, since it shows both the overall network structure and fine-grain visualization in the most detailed level, automatically placing semantically similar metabolites next to each other. Mimoza does not depend on pathway information, automatically infers the relative compartment placement (e.g. places organelles inside the cytoplasm) and exploits a model in SBML format with ChEBI annotations for metabolites (if no annotations are present, it tries to infer them automatically based on metabolites' names).

Using generalization to compare two metabolic networks makes most sense if they have equivalent generalized nodes that can be placed in corresponding positions in the two layouts. Mimoza currently handles this correspondence between zoom levels of the same network, but does not guarantee such correspondence when two networks are laid out independently. To meet this challenge, three strategies can be explored. The first is to use constrained layout [Böhringer and Paulisch, 1990], to impose the positions of key features in one network on the corresponding features of the second network. The second is also to use constrained layout, with a catalog of standard positions for common motifs in generalized maps; for example, always lay out the generalized β-oxidation

of fatty acids as a 4-step cycle, with standard positions for the generalized metabolites common for all the networks that incorporate β-oxidation. The third strategy, which we are in the process of testing, is to learn a common layout by generalizing the union of the two networks. The idea is to combine the reactions into one set, run the generalization procedure on the union to fix the positions of the common features, then to build each of the layouts using only its own set of nodes. Each network layout only contains its own nodes, but the common nodes of the two networks will be in common positions.

Finally, the API of the Leaflet framework used for the interactive navigation can be used to integrate the maps with other web-based tools, such as annotation editors or simulation software.

Mimoza is currently targeted to metabolic networks. While it can provide a geometric zooming visualization of a generic SBML model (e.g., a signaling network), the knowledge-based generalization, and therefore semantic zooming, depends on the ChEBI ontology and is intended for metabolic models. A domain-specific adaptation of the generalization method (e.g., use of a domain-specific ontology instead of ChEBI, that is targeted to metabolism) might allow Mimoza to assist in modeling of other kinds of biological networks.

5.4 Conclusions

We have implemented Mimoza, a novel software tool for automatically constructing zooming user interfaces for genome-scale metabolic models. By exploiting *model generalization*, Mimoza reduces the dimension of the model's network at outer zoom levels, and intelligently co-localizes equivalent reactions and molecular species at inner zoom levels. Consequently the biological user may efficiently navigate the high-level structure of the model; whether the goal is to understand the model or to search for errors, Mimoza exposes the important features at out zoom levels and and hides the specific details in the inner ones. We provide an efficient, useful tool that is easy to adopt and, through the use of standards such as SBML and the ChEBI ontology, is easy to integrate into existing expert-centered modeling pipelines. By carefully combining model generalization with adaptive layout and open-source cartographic software, the Mimoza web server requires just a browser with Javascript. Mimoza is open source and can also be installed locally, as described on the web page, and depends on libSBML, Tulip, and Python.

5.5 Availability and requirements

Project name: Mimoza
Project home page: http://mimoza.bordeaux.inria.fr
Operating system(s): Platform independent
Programming language: Python, JavaScript
Other requirements: JavaScript should be enabled in the web browser. The standalone Mimoza application requires Python 2.7; libSBML-experimental ≥ 5.9 for Python with groups and layout extensions; Leaflet 0.7.3; jQuery 2.1.1 and jQuery-ui 1.10.4; Tulip ≥ 4.0 for python; and model generalization library[1].
License: CeCILL (GPL compatible)
Any restrictions to use by non-academics: no restrictions

[1] Model Generalization – http://metamogen.gforge.inria.fr

Table 5.1 Comparison of ZUIs for metabolic models.

Tool name	Fixed layout	Semantic zoom	User's model	Automatic layout	Modules
Genome Projector	yes	no	no	-	no
NaviCell	no	if created by user	yes	no	yes
Cellular Overview	yes	no	no	-	no
Reactome	yes (same pathw. diff. org.) / no (diff. pathw. same organism)	yes	no	-	yes
Mimoza	no	yes	yes	yes	yes

Chapter 6

Conclusions

6.1 Main contributions

The complexity of large-scale metabolic networks makes their analysis and curation hard for a human expert. The abundance of details, needed for a computer simulation, may hide errors and particularities, that require curator's attention. Finding the right level of abstraction that allows a human expert to study the structure of the network and draws curator's attention to networks' specificities is important during model creation, comparison and knowledge-based exploration.

To address this issue, we defined a knowledge-based generalization that allows for production of higher-level abstract views of metabolic network models. The generalized views preserve essential model structure and highlight the particularities.

To perform the model generalization, we developed a theoretical method that groups similar metabolites and reactions in the network based on its structure and the knowledge extracted from metabolite ontologies, and then compresses the model based on this grouping. The generalization of metabolic networks is possible due to the following three key properties.

1. A metabolic network can be regarded as a *bipartite* graph [Diestel, 2012] with two disjoint sets of nodes (metabolites and reactions), and edges that connect the reactions to their substrate and product metabolites. The bipartite nature of the graph is important for the generalization method, as the generalization is performed differently for the two types of nodes, and for reaction nodes depends on the fact that the edges connect them only to metabolites.

2. The association between metabolites and the ChEBI ontology terms is another key property needed for the generalization. The ChEBI hierarchy defines a *partial or-*

der of its terms. Every metabolite can be generalized up to one of its ancestors depending on the model structure.

3. The last necessary property is the *repetitive structure* of the graph. Only reactions that have the same numbers of substrates and of products (i.e., nodes with the same in- and out-degrees) are considered as candidates for generalization; these numbers should be conserved also after the reaction factoring. Thus, repetitive patterns in the graph structure are needed for the generalization to be efficient.

Overall, the generalization method is currently composed of three modules:

1. *Aggressive reaction grouping* based on the most general metabolite grouping, in order to generate reaction grouping candidates;

2. *Ungrouping of some metabolites and reactions* to correct for violation of the stoichiometry preserving constraint;

3. *Ungrouping of some metabolites* (while keeping the reaction grouping intact) to correct for violation of the metabolite diversity constraint.

The reaction *stoichiometry preserving constraint* is crucial for finding an appropriate metabolite generalization level. In graph terms, this constraint imposes that the in- and out-degrees of the reaction nodes must not be changed by the generalization. Currently, during a model generalization, this constraint is satisfied separately for each group of reactions, but the result is propagated on the whole model: To satisfy the stoichiometry constraint the ancestor of a group of metabolites that causes the conflict is replaced by several more specific ancestors, thus, splitting the metabolite group, and consequently the reaction one. The effect of the generalized metabolite partition is *model-wide*, i.e., if the same generalized metabolite participated in another reaction group, even without violating its stoichiometry, its partition may cause the partition of that reaction group as well. By using the model-wide stoichiometry constraint we achieve a metabolite level that is consistent with the structures of reactions in the model.

The *metabolite diversity constraint* imposes that the metabolite grouping is supported by the reaction factoring: The ancestors for metabolites are chosen as the most specific ones permitting the found reaction factoring. Hence, in graph terms, the generalization seeks to minimize the degrees of the generalized metabolite nodes, given that the reaction grouping is already calculated.

We implemented the method as a python library, that is available for download from metamogen.gforge.inria.fr. To validate our method we applied it to 1 286 metabolic mod-

els from the Path2Model project, and showed that it helps to detect organism-, and domain-specific adaptations and to compare the models.

Based on discussions with users and their ways of navigation in metabolic networks, we chose a 3-level representation of metabolic models: the compartment level, the generalized level (obtained with our generalization method), the full-model level. We developed Mimoza, a user-centric tool for zoomable navigation and knowledge-based exploration of metabolic networks that produces this representation. The 3-level representation allows for analysis of metabolic models in a top-down manner, starting from general question about model compartments, continuing with the verification of the generalized model structure, and finally checking the details of the complete model, needed for simulation. Mimoza is available both as an on-line tool and for download at mimoza.bordeaux.inria.fr.

6.2 Perspectives

The contributions of the the thesis lead to a number of interesting perspectives that explore the compression, comparison and classification capabilities of model generalization.

6.2.1 Compressing bipartite graphs with repetitions

As we have seen, our generalization method is efficient on metabolic networks that contain *repetitive patterns*, i.e., reactions of similar structure, which operate on similar substrates and products. The ChEBI ontology allows for grouping of related metabolites up to a certain level of abstraction, while self-similarities in the topology of the network graph allow for reaction grouping.

It is noteworthy that the generalization algorithm does not depend on the metabolic origin of the network, and could be potentially applied to any graph that contains repetitive patterns in its structure if there exists a way to bring some of its nodes to common levels of abstraction (e.g., a node ontology).

To define it in a more formal way, the generalization method can be applied to a *bipartite graph* for which a *partial order* (e.g., an ontology) is defined on one type of its nodes (e.g., metabolites) to infer a grouping of nodes of the second type (e.g., reactions) and to compress the graph based on this grouping. The generalization detects *repetitive patterns* in the graph and factors them to obtain a *compressed graph*, containing one representative element per pattern. The generalization preserves the in- and out-degrees of

the nodes of the second type (reaction stoichiometries in case of metabolic graphs), and minimizes the degrees of the nodes of the first type (satisfies metabolite diversity constraint in case of metabolic graphs) in the compressed graph. The compressed graph is an approximation of a graph with the minimal number of nodes of the second type.

6.2.2 Finding reference models for model inference

As we have just seen the generalization method compresses the knowledge stored in a graph. In the case of metabolic network graphs this compressed representation can serve as a reference for model inference.

Among the various metabolic model inference methods, the most useful for our group is the one of the Pantograph toolbox [Loira et al., 2014]. To produce a draft model for a target species, it uses a model for a related organism as a template and combines several sources of orthology between reference and target species' genomes to define which of the template's reactions should be conserved in the target model and to update their gene associations.

The reference model serves as a knowledge base for a metabolic reconstruction. It is thus important to find the right compromise between the details and generality: A too-detailed and organism-specific model would include very substrate- and organism-specific gene associations and thus complicate gene rewriting; a too generic one might not cover reactions specific to this group of organisms. Generalized models are good candidates for model inference templates. They bring reactions to less substrate-specific levels, while keeping the organism-specific adaptations and alternative paths.

Moreover, if models of several close species exist, a *collective generalization* of those models could serve as an even better template, which is not biased towards a particular reference metabolism, and distinguishes the conserved common part of the group's metabolisms from the organism-specific adaptations. By a collective generalization of several models we understand a generalization of a collection of all the reactions and metabolites found in those models. The generalized model would group together the conserved parts and include model-specific particularities. A collective generalization can also be used for merging partial models, for example, those describing a particular pathway or a metabolic sub-system.

Using generalized models will however require an update of the Pantograph method. Currently, each reaction added to the initial draft model corresponds to exactly one reaction in the reference model (with the rewritten gene associations, but the same reactant and product metabolites). If the generalized model is used as a template instead, a conserved reaction might need to be specified in the target model, producing several target

reactions operating on similar, more specific, metabolites.

A collective generalization using a flat metabolite ontology (without hierarchical relationships), would perform a standard model merge, removing the duplicates from the combined model. Such a merged model can be used as a template in the current Pantograph method, not requiring its modifications. As in several other model merging methods [Coskun et al., 2013; Krause et al., 2010], our one makes use of ontological identifiers to detect model intersections; what makes it different is the fact that it stores the numbers of reactions factored together during the merge. This information shows which parts were conserved between the models, and could be potentially used as weights in the Pantograph method: A reaction with a greater weight (better conserved in the models used for the template) is more probable to be conserved also in the target model.

6.2.3 Comparing disease and healthy metabolisms

In aforementioned case of model inference the generalization was used to detect the part of metabolism that is *conserved* between species. The complement thing that generalization allows one to do is identification of the *metabolic differences* with respect to the common part. The situation where it is especially important is comparison of a model for a healthy metabolism to a one suffering from a disease. It is even more informative if the models for several organisms affected by the disease are available: A collective generalization, which we proposed before as reference models for model inference, can be applied to the disease-affected models and compared to a collective generalization of healthy ones. It would permit detection of the common, disease-specific, adaptation of their metabolisms, as well as of the conserved part that is not affected by the disease.

Another direction is to investigate and define *generalized disease-related differences* between the models. If both a model of the healthy and a model of the disease-affected metabolisms are available, the difference between the (potentially generalized) models can be computed. Examples of non-generalized disease-related differences can be found in KEGG DISEASE database [Kanehisa, 2009], which contains pathway maps for cancer, immune disorders, neurodegenerative diseases, etc. in human. It would be, even more so, interesting to find a generalized disease-related difference that is not bound to a particular organism, and study if when applied to a model of a healthy metabolism it could produce a draft model of metabolism suffering from this disease.

6.2.4 Classifying related metabolisms

As we have just seen comparison of models can benefit from the generalization. Indeed, it permits one to detect well-conserved model structure and highlights model-specific particularities, no matter whether the compared models describe the same organism in different states (e.g., healthy and affected by a disease) or different but related organisms. Taken to a multi-organism scale, model generalization can therefore be used in *metabolic taxonomy* [Hong et al., 2004].

Taxonomy (systematics) is the science of biological classification. It consists of three main activities: recognition of species, classification into a hierarchical scheme, and placing the information about species and their classification in a broader context [Schuh, 2000]. Metabolic taxonomy classifies species based on metabolic traits (e.g., based on substrate-product relationships [Chang et al., 2011], metabolic pathways [Hong et al., 2004; Mazurie et al., 2008] or enzyme information [Ma and Zeng, 2004]) as opposite to more traditional genomic methods, which study mutations in the sequence of orthologous genes found in all the species of interest [Olsen et al., 1994].

A collective generalization of a group of models of related organisms would allow one to detect the conserved part of metabolism, common to all of them (i.e., common metabolic ancestor). The closer metabolisms of the organisms are, the larger the conserved part is. Comparison of individual generalized models to this collective generalization can be used to find organism-specific features and adaptations, in order to study the connection between the differences in generalized metabolic properties and the differences in organisms' physiology.

6.2.5 Classifying reactions in reaction databases

Generalization can be used for classification not only on the organism level, but also on the reaction one. There exist several databases storing metabolic reactions, including Rhea [Alcántara et al., 2012], BioPath [Reitz et al., 2004], KEGG reaction database [Kanehisa et al., 2012] and MetaCyC [Caspi et al., 2012]. They represent such information about reactions as involved metabolites with links to metabolite databases, reversibility, catalyzing enzymes, pathways in which they participate, cross references, etc. Of these only Rhea classifies reactions, and tries to link reactions to similar ones.

Rhea is a manually annotated database of chemical reactions. Along with transport and spontaneous reactions, it covers the official list of enzyme-catalyzed reactions defined by the Nomenclature Committee of the IUBMB (http://www.chem.qmul.ac.uk/iubmb/enzyme/), which implies the classification based on Enzyme Commission (EC)

numbers. Rhea includes so called generic reactions, that are catalyzed by enzymes with broad substrate specificity, e.g., *RHEA:10739*: a primary alcohol + NAD^+ ↔ an aldehyde + H^+ + NADH, catalyzed by alcohol dehydrogenase (EC 1.1.1.1). Rhea also contains more precise reactions for known specific substrate/product pairs. However, Rhea does not provide an explicit link between the generic and specific reactions.

The metabolites in Rhea are associated with the entities in ChEBI, which makes Rhea reactions compatible with our generalization method. The generic Rhea reactions can be viewed as analogues of the generalized reactions defined in this thesis. However, the origins of these reactions are different: Generalized reactions created by our method are based on metabolite hierarchy in ChEBI coupled with the constrains imposed by the metabolic model of interest, while generic reactions in Rhea are derived from the IUBMB classification, are not organism-specific and might not explore all the possibilities.

Applying our method to RhEA could allow for structuring reactions hierarchically and providing organism-specific hierarchical views with organism-specific generic reactions: The level in ChEBI to which a metabolite can be generalized depends on the constraints imposed by a metabolic model, and may differ from one organism to another; this in turn influences the reaction generalization.

To build a reaction hierarchy for a reaction database, our method can be applied in three following ways.

1. Regarding the whole reaction database as one metabolic model, we can generalize it using database-wide stoichiometry constraints. Assuming that the database includes all the currently known metabolic reactions, this generalization will be the most specific one, compatible with stoichiometric constraints imposed by any possible metabolic model. After satisfying the metabolite diversity constraint, this generalization will define the direct ancestors for reactions in the database.

2. Detection of similar reactions on the whole database followed by group-wide stoichiometry preserving, as if every reaction group formed an independent model, would create the most general reaction groupings. After satisfying the metabolite diversity constraints, this generalization will define the root ancestors for reactions in the database.

3. Running a generalization on a subset of the database reactions that are found in a particular model, i.e., a model-wide generalization, would produce intermediate ancestors for those reactions. These ancestors would be compatible with the specific model.

We are currently working on the first two generalizations for the Rhea database.

6.2.6 Suggesting extensions to metabolite ontologies

The model generalization method relies on the metabolite classification provided by a metabolite ontology (ChEBI) and, as we have shown, can be used to classify reactions and even organisms. But our method could be extended to also predict the relationships between metabolites themselves.

Currently, the method cannot always use the ChEBI relationships between metabolites, as for some metabolites their ChEBI annotations are not included in the model and cannot be easily deduced; moreover, for some metabolites corresponding terms do not yet exist in ChEBI, like for several metabolites in the model of the global reconstruction of human metabolism [Thiele et al., 2013].

For metabolite grouping the model generalization method relies on their relationships in ChEBI (to group only related metabolites, which have a common ancestor), and also on their participation in similar reactions. For metabolites unknown to ChEBI, it would be, therefore, interesting to define a *relaxed generalization method* based only on the presence of similar reactions. Of course, the stoichiometry and metabolite diversity constraints must be still satisfied.

The groups to which the metabolites unknown to ChEBI are assigned by the relaxed generalization, could suggest their potential place in the ChEBI hierarchy: If the group contains also ChEBI-annotated metabolites, their common ancestor can be suggested as a potential ancestor for the new terms. This ancestor proposal is supported by its consistency with the model structure.

To perform the relaxed generalization the method will need to be modified in the following way. Currently, each reaction is assigned a key, defined by the generalizations of its reactants and products: ancestor ChEBI identifiers for generalized metabolites or their own identifiers for ubiquitous metabolites and metabolites that are not found in ChEBI. The reaction grouping is performed by matching these keys. The relaxed generalization method will include another iteration of the generalization, using a *fuzzy* reaction key: strict matching of ubiquitous participants plus matching of the numbers of specific reactants and products. The generalization found by the fuzzy reaction grouping should be further updated to satisfy the constraints.

References

Agafonkin, V. 2010. Leaflet - a JavaScript library for mobile-friendly maps. URL http://leafletjs.com/.

Agren, R., Liu, L., Shoaie, S., et al. 2013. The RAVEN toolbox and its use for generating a genome-scale metabolic model for Penicillium chrysogenum. *PLoS computational biology* 9(3), e1002980.

Alberts, B., Johnson, A., Lewis, J., et al. 2007. *Molecular Biology of the Cell*. Taylor & Francis Group.

Alcántara, R., Axelsen, K.B., Morgat, A., et al. 2012. Rhea–a manually curated resource of biochemical reactions. *Nucleic Acids Research* 40(Database issue), D754–60.

Arakawa, K., Tamaki, S., Kono, N., et al. 2009. Genome Projector: zoomable genome map with multiple views. *BMC bioinformatics* 10(1), 31.

Ashburner, M., Ball, C.A., Blake, J.A., et al. 2000. Gene Ontology: tool for the unification of biology. *Nature Genetics* 25(1), 25–29.

Auber, D. 2004. Tulip — A Huge Graph Visualization Framework. *In* M. Jünger, P. Mutzel, G. Farin, H.C. Hege, D. Hoffman, C.R. Johnson, K. Polthier, and M. Rumpf, eds., *Graph Drawing Software*, Mathematics and Visualization, 105–126. Springer Berlin Heidelberg, Berlin Heidelberg.

Aung, H.W., Henry, S.A., and Walker, L.P. 2013. Revising the Representation of Fatty Acid, Glycerolipid, and Glycerophospholipid Metabolism in the Consensus Model of Yeast Metabolism. *Industrial biotechnology (New Rochelle, N.Y.)* 9(4), 215–228.

Aziz, R.K., Bartels, D., Best, A.A., et al. 2008. The RAST Server: rapid annotations using subsystems technology. *BMC genomics* 9, 75.

Barillot, E., Guyon, F., Cussat-Blanc, C., et al. 1998. HuGeMap: a distributed and integrated Human Genome Map database. *Nucleic acids research* 26(1), 106–7.

Bederson, B. and Meyer, J. 1998. Implementing a zooming User Interface: experience building Pad++. *Software: Practice and Experience* 28(10), 1101–1135.

Benson, D.A., Clark, K., Karsch-Mizrachi, I., et al. 2014. GenBank. *Nucleic Acids Research* .

Bergmann, F.T., Adams, R., Moodie, S., et al. 2014. One file to share them all: Using the COMBINE Archive and the OMEX format to share all information about a modeling project .

Bhasker, J. and Samad, T. 1991. The clique-partitioning problem. *Computers & Mathematics with Applications* 22(6), 1–11.

Blankenberg, D., Von Kuster, G., Coraor, N., et al. 2010. Galaxy: a web-based genome analysis tool for experimentalists. *Curr Protoc Mol Biol* Chapter 19, Unit 19.10.1–21.

Boele, J., Olivier, B.G., and Teusink, B. 2012. FAME, the Flux Analysis and Modeling Environment. *BMC systems biology* 6(1), 8.

Böhringer, K.F. and Paulisch, F.N. 1990. Using constraints to achieve stability in automatic graph layout algorithms. *In Proceedings of the SIGCHI conference on Human factors in computing systems Empowering people - CHI '90*, 43–51. ACM Press, New York, New York, USA.

Bonarius, H.P., Schmid, G., and Tramper, J. 1997. Flux analysis of underdetermined metabolic networks: the quest for the missing constraints. *Trends in Biotechnology* 15(8), 308–314.

Bornstein, B.J., Keating, S.M., Jouraku, A., et al. 2008. LibSBML: an API library for SBML. *Bioinformatics* 24(6), 880–1.

Büchel, F., Rodriguez, N., Swainston, N., et al. 2013. Path2Models: large-scale generation of computational models from biochemical pathway maps. *BMC systems biology* 7(1), 116.

Butler, H., Daly, M., Doyle, A., et al. 2008. GeoJSON Specification. URL http://geojson.org/geojson-spec.html.

Caspi, R., Altman, T., Dreher, K., et al. 2012. The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic acids research* 40(Database issue), D742–53.

Chance, B. 1943. The kinetics of the enzyme-substrate compound of peroxidase. *J Biol Chem* 151, 553–577.

Chang, C.W., Lyu, P.C., and Arita, M. 2011. Reconstructing phylogeny from metabolic substrate-product relationships. *BMC bioinformatics* 12 Suppl 1, S27.

Chavali, A.K., D'Auria, K.M., Hewlett, E.L., et al. 2012. A metabolic network approach for the identification and prioritization of antimicrobial drug targets. *Trends in microbiology* 20(3), 113–23.

Chibucos, M.C., Mungall, C.J., Balakrishnan, R., et al. 2014. Standardized description of scientific evidence using the Evidence Ontology (ECO). *Database : the journal of biological databases and curation* 2014.

Chvatal, V. 1979. A Greedy Heuristic for the Set-Covering Problem. *Mathematics of Operations Research* 4(3), 233–235.

Clugston, M. and Flemming, R. 2000. *Advanced Chemistry (Advanced Science)*. OUP Oxford.

Copeland, W.B., Bartley, B.A., Chandran, D., et al. 2012. Computational tools for metabolic engineering. *Metabolic Engineering* 14(3), 270–280.

Coskun, S.A., Cicek, A.E., Lai, N., et al. 2013. An online model composition tool for system biology models. *BMC systems biology* 7, 88.

Courtot, M., Juty, N., Knüpfer, C., et al. 2011. Controlled vocabularies and semantics in systems biology. *Molecular systems biology* 7, 543.

Croft, D. 2013. Building models using Reactome pathways as templates. *Methods in Molecular Biology* 1021, 273–83.

de Matos, P., Alcántara, R., Dekker, A., et al. 2010. Chemical Entities of Biological Interest: an update. *Nucleic Acids Research* 38(suppl 1), D249–D254.

Demir, E., Cary, M.P., Paley, S., et al. 2010. The BioPAX community standard for pathway data sharing. *Nature biotechnology* 28(9), 935–42.

Devoid, S., Overbeek, R., DeJongh, M., et al. 2013. Automated genome annotation and metabolic model reconstruction in the SEED and Model SEED. *Methods in molecular biology (Clifton, N.J.)* 985, 17–45.

Diestel, R. 2012. *Graph Theory: Springer Graduate Text GTM 173*. Reinhard Diestel.

Ebrahim, A., Lerman, J.A., Palsson, B.O., et al. 2013. COBRApy: COnstraints-Based Reconstruction and Analysis for Python. *BMC systems biology* 7(1), 74.

Edwards, J.S., Covert, M., and Palsson, B. 2002. Metabolic modelling of microbes: the flux-balance approach. *Environmental microbiology* 4(3), 133–40.

Edwards, J.S. and Palsson, B.O. 1999. Systems Properties of the Haemophilus influenzae Rd Metabolic Genotype. *Journal of Biological Chemistry* 274(25), 17410–17416.

Edwards, J.S. and Palsson, B.O. 2000. The Escherichia coli MG1655 in silico metabolic genotype: Its definition, characteristics, and capabilities. *Proceedings of the National Academy of Sciences* 97(10), 5528–5533.

Falb, M., Müller, K., Königsmaier, L., et al. 2008. Metabolism of halophilic archaea. *Extremophiles : life under extreme conditions* 12(2), 177–96.

Fang, K., Zhao, H., Sun, C., et al. 2011. Exploring the metabolic network of the epidemic pathogen Burkholderia cenocepacia J2315 via genome-scale reconstruction. *BMC systems biology* 5, 83.

Feige, U. 1998. A threshold of ln n for approximating set cover. *Journal of the ACM* 45(4), 634–652.

Finney, A. and Hucka, M. 2003. Systems Biology Markup Language (SBML) Level 2: Structures and Facilities for Model Definitions. Tech. rep. URL http://co.mbine.org/specifications/sbml.level-2.version-1.pdf.

Finney, A., Hucka, M., and Le Novère, N. 2006. Systems Biology Markup Language (SBML) Level 2: Structures and Facilities for Model Definitions. Tech. rep. URL http://co.mbine.org/specifications/sbml.level-2.version-2.pdf.

Fleet, G.H. 2007. Yeasts in foods and beverages: impact on product quality and safety. *Current opinion in biotechnology* 18(2), 170–5.

Fleischmann, R.D., Adams, M.D., White, O., et al. 1995. Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. *Science (New York, N.Y.)* 269(5223), 496–512.

Förster, J., Famili, I., Fu, P., et al. 2003. Genome-scale reconstruction of the Saccharomyces cerevisiae metabolic network. *Genome research* 13(2), 244–53.

Fruchterman, T.M.J. and Reingold, E.M. 1991. Graph drawing by force-directed placement. *Software: Practice and Experience* 21(11), 1129–1164.

Funahashi, A., Matsuoka, Y., Jouraku, A., et al. 2008. CellDesigner 3.5: A Versatile Modeling Tool for Biochemical Networks. *Proceedings of the IEEE* 96(8), 1254–1265.

Gasteiger, E., Gattiker, A., Hoogland, C., et al. 2003. ExPASy: The proteomics server for in-depth protein knowledge and analysis. *Nucleic acids research* 31(13), 3784–8.

Gauges, R., Rost, U., Sahle, S., et al. 2013. SBML Level 3 Layout Package Version 1 Release 1. URL http://identifiers.org/combine.specifications/sbml.level-3.version-1.layout.version-1.release-1.

Goldreich, O. 2008. *Computational Complexity: A Conceptual Perspective*. Cambridge University Press, Cambridge.

Hachul, S. and Jünger, M. 2005. Large-graph layout with the fast multipole multilevel method. Tech. rep., Universität zu Köln, Institut für Informatik, Köln. URL http://e-archive.informatik.uni-koeln.de/509/2/zaik2006-509.pdf.

Hamilton, J.J. and Reed, J.L. 2014. Software platforms to facilitate reconstructing genome-scale metabolic networks. *Environmental microbiology* 16(1), 49–59.

Herman, I., Melancon, G., and Marshall, M. 2000. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on Visualization and Computer Graphics* 6(1), 24–43.

Herrgå rd, M.J., Swainston, N., Dobson, P., et al. 2008. A consensus yeast metabolic network reconstruction obtained from a community approach to systems biology. *Nature biotechnology* 26(10), 1155–60.

Hollinshead, W., He, L., and Tang, Y.J. 2014. Biofuel production: an odyssey from metabolic engineering to fermentation scale-up. *Frontiers in Microbiology* 5, 344.

Hong, S.H., Kim, T.Y., and Lee, S.Y. 2004. Phylogenetic analysis based on genome-scale metabolic pathway reaction content. *Applied microbiology and biotechnology* 65(2), 203–10.

Hoops, S., Sahle, S., Gauges, R., et al. 2006. COPASI, a COmplex PAthway SImulator. *Bioinformatics* 22, 3067–3074.

Hu, Z., Mellor, J., Wu, J., et al. 2007. Towards zoomable multidimensional maps of the cell. *Nature biotechnology* 25(5), 547–54.

Hucka, M. 2012. Groups Proposal. URL http://sbml.org/Community/Wiki/SBML_Level_3_Proposals/Groups_Proposal_Updated_%282012-06%29.

Hucka, M., Bergmann, F.T., Hoops, S., et al. 2010. The Systems Biology Markup Language (SBML): Language Specification for Level 3 Version 1 Core. Tech. rep. URL http://co.mbine.org/specifications/sbml.level-3.version-1.core.release-1.pdf.

Hucka, M., Finney, A., Sauro, H., et al. 2001. Systems Biology Markup Language (SBML) Level 1: Structures and Facilities for Basic Model Definitions. Tech. rep. URL http://co.mbine.org/specifications/sbml.level-1.version-1.pdf.

Hucka, M., Finney, A., Sauro, H.M., et al. 2003. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics (Oxford, England)* 19(4), 524–31.

Jensen, P.A. and Papin, J.A. 2014. MetDraw: automated visualization of genome-scale metabolic network reconstructions and high-throughput data. *Bioinformatics (Oxford, England)* .

Jianu, R. and Laidlaw, D.H. 2013. What Google Maps can do for biomedical data dissemination: examples and a design study. *BMC research notes* 6(1), 179.

Juty, N., Le Novère, N., and Laibe, C. 2012. Identifiers.org and MIRIAM Registry: community resources to provide persistent identification. *Nucleic acids research* 40(Database issue), D580–6.

Kanehisa, M. 2009. Representation and analysis of molecular networks involving diseases and drugs. *Genome informatics. International Conference on Genome Informatics* 23(1), 212–3.

Kanehisa, M., Goto, S., Sato, Y., et al. 2012. KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Res.* 40(0305-1048 (Linking)), D109–14.

Karp, P.D., Paley, S., and Romero, P. 2002. The Pathway Tools software. *Bioinformatics* 18(Suppl 1), S225–S232.

Karp, R.M. 1972. Reducibility Among Combinatorial Problems. *In* R.E. Miller and J.W. Thatcher, eds., *Complexity of Computer Computations*, 85–103. Plenum Press.

Karr, J.R., Sanghvi, J.C., Macklin, D.N., et al. 2012. A whole-cell computational model predicts phenotype from genotype. *Cell* 150(2), 389–401.

Kim, H.U., Kim, T.Y., and Lee, S.Y. 2010. Genome-scale metabolic network analysis and drug targeting of multi-drug resistant pathogen Acinetobacter baumannii AYE. *Molecular bioSystems* 6(2), 339–48.

Kim, T.Y., Sohn, S.B., Kim, Y.B., et al. 2012. Recent advances in reconstruction and applications of genome-scale metabolic models. *Current opinion in biotechnology* 23(4), 617–23.

Klamt, S., Saez-Rodriguez, J., and Gilles, E.D. 2007. Structural and functional analysis of cellular networks with CellNetAnalyzer. *BMC Systems Biology* 1.

Köhn, D. and Le Novère, N. 2008. SED-ML - An XML Format for the Implementation of the MIASE Guidelines. *In* M. Heiner and A. Uhrmacher, eds., *CMSB '08 Proceedings of the 6th International Conference on Computational Methods in Systems Biology*, vol. 5307 of *Lecture Notes in Computer Science*, 176–190. Springer Berlin / Heidelberg.

Kohonen, T. 1982. Self-organized formation of topologically correct feature maps. *Biological Cybernetics* 43(1), 59–69.

Krause, F., Uhlendorf, J., Lubitz, T., et al. 2010. Annotation and merging of SBML models with semanticSBML. *Bioinformatics (Oxford, England)* 26(3), 421–2.

Kuperstein, I., Cohen, D.P., Pook, S., et al. 2013. NaviCell: a web-based environment for navigation, curation and maintenance of large molecular interaction maps. *BMC Systems Biology* 7(1), 100.

Latendresse, M. and Karp, P.D. 2011. Web-based metabolic network visualization with a zooming user interface. *BMC Bioinformatics* 12(1), 176.

Laufer, L., Halacsy, P., and Somlai-Fischer, A. 2011. Prezi meeting: collaboration in a zoomable canvas based environment. *In Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems - CHI EA '11*, 749. ACM Press, New York, New York, USA.

Le Novere, N., Demir, E., Mi, H., et al. 2011. Systems Biology Graphical Notation: Entity Relationship language Level 1 (Version 1.2). *Nature Precedings* .

Le Novère, N., Hucka, M., Mi, H., et al. 2009a. The Systems Biology Graphical Notation. *Nature Biotechnology* 27(8), 735–41.

Le Novère, N., Hucka, M., Mi, H., et al. 2009b. The Systems Biology Graphical Notation. *Nature Biotechnology* 27(8), 735–41.

Li, C., Donizelli, M., Rodriguez, N., et al. 2010. BioModels Database: An enhanced, curated and annotated resource for published quantitative kinetic models. *BMC Systems Biology* 4, 92.

Liu, Y., Zhu, Y., Li, J., et al. 2014. Modular pathway engineering of Bacillus subtilis for improved N-acetylglucosamine production. *Metabolic engineering* 23, 42–52.

Lloyd, C.M., Halstead, M.D.B., and Nielsen, P.F. 2004. CellML: its future, present and past. *Progress in biophysics and molecular biology* 85(2-3), 433–50.

Loira, N., Dulermo, T., Nicaud, J.M., et al. 2012. A genome-scale metabolic model of the lipid-accumulating yeast Yarrowia lipolytica. *BMC Systems Biology* 6(1), 35.

Loira, N., Zhukova, A., and Sherman, D.J. 2014. Pantograph: A template-based method for genome-scale metabolic model reconstruction. *Journal of bioinformatics and computational biology* 1550006.

Ma, H.W. and Zeng, A.P. 2004. Phylogenetic comparison of metabolic capacities of organisms at genome level. *Molecular phylogenetics and evolution* 31(1), 204–13.

Mazurie, A., Bonchev, D., Schwikowski, B., et al. 2008. Phylogenetic distances are encoded in networks of interacting pathways. *Bioinformatics (Oxford, England)* 24(22), 2579–85.

McGuinness, D.L. and van Harmelen, F. 2004. OWL Web Ontology Language Overview. URL http://www.w3.org/TR/owl-features/.

Metzler, D.E. and Metzler, C.M. 2001. *Biochemistry: The Chemical Reactions of Living Cells, 2nd Edition*. No. v. 1 in Biochemistry: The Chemical Reactions of Living Cells, 2nd ed. Academic Press, San Diego.

Mi, H., Schreiber, F., Le Novére, N., et al. 2009. Systems Biology Graphical Notation: Activity Flow language Level 1. *Nature Precedings* .

Milacic, M., Haw, R., Rothfels, K., et al. 2012. Annotating Cancer Variants and Anti-Cancer Therapeutics in Reactome. *Cancers* 4(4), 1180–1211.

Moodie, S., Le Novere, N., Demir, E., et al. 2011. Systems Biology Graphical Notation: Process Description language Level 1. *Nature Precedings* .

Muto, A., Kotera, M., Tokimatsu, T., et al. 2013. Modular Architecture of Metabolic Pathways Revealed by Conserved Sequences of Reactions. *Journal of chemical information and modeling* 53(3), 613–22.

NCBI 2014. NCBI Gene. URL http://www.ncbi.nlm.nih.gov/gene.

Nivala, A.M., Brewster, S., and Sarjakoski, T.L. 2008. Usability Evaluation of Web Mapping Sites. *The Cartographic Journal* 45(2), 129–138.

Olsen, G.J., Woese, C.R., and Overbeek, R. 1994. The winds of (evolutionary) change: breathing new life into microbiology. *Journal of bacteriology* 176(1), 1–6.

Orth, J.D., Conrad, T.M., Na, J., et al. 2011. A comprehensive genome-scale reconstruction of Escherichia coli metabolism–2011. *Molecular Systems Biology* 7(1), 535.

Orth, J.D., Thiele, I., and Palsson, B.O. 2010. What is flux balance analysis? *Nature biotechnology* 28(3), 245–8.

Palsson, B. 2006. *Systems biology properties of reconstructed networks*. Cambridge University Press, Cambridge, New York.

Palsson, B.O. 2011. *Systems biology simulation dynamic network states*. Cambridge University Press, Cambridge, New York.

Pitkänen, E., Jouhten, P., Hou, J., et al. 2014. Comparative Genome-Scale Reconstruction of Gapless Metabolic Networks for Present and Ancestral Species. *PLoS Computational Biology* 10(2), e1003465.

Pook, S., Vaysseix, G., and Barillot, E. 1998. Zomit: biological data visualization and browsing. *Bioinformatics* 14(9), 807–14.

Reed, J.L. and Palsson, B.O. 2003. Thirteen years of building constraint-based in silico models of Escherichia coli. *Journal of bacteriology* 185(9), 2692–9.

Reitz, M., Sacher, O., Tarkhov, A., et al. 2004. Enabling the exploration of biochemical pathways. *Organic & biomolecular chemistry* 2(22), 3226–37.

Roberts, J.C. 2005. *Exploratory Visualization with Multiple Linked Views*. Elsevier, Pergamon.

Rohn, H., Junker, A., Hartmann, A., et al. 2012. VANTED v2: a framework for systems biology applications. *BMC systems biology* 6(1), 139.

Rubin, D.L., Shah, N.H., and Noy, N.F. 2008. Biomedical ontologies: a functional perspective. *Briefings in bioinformatics* 9(1), 75–90.

Sayers, E.W., Barrett, T., Benson, D.A., et al. 2009. Database resources of the National Center for Biotechnology Information. *Nucleic acids research* 37(Database issue), D5–15.

Schellenberger, J., Park, J.O., Conrad, T.M., et al. 2010. BiGG: a Biochemical Genetic and Genomic knowledgebase of large scale metabolic reconstructions. *BMC bioinformatics* 11, 213.

Schilling, C.H., Covert, M.W., Famili, I., et al. 2002. Genome-Scale Metabolic Model of Helicobacter pylori 26695. *Journal of Bacteriology* 184(16), 4582–4593.

Schilling, C.H. and Palsson, B.O. 2000. Assessment of the metabolic capabilities of Haemophilus influenzae Rd through a genome-scale pathway analysis. *Journal of theoretical biology* 203(3), 249–83.

Schuh, R.T. 2000. *Biological Systematics: Principles and Applications*. Cornell University Press.

Schulz, M., Uhlendorf, J., Klipp, E., et al. 2006. SBMLmerge, a system for combining biochemical network models. *Genome informatics. International Conference on Genome Informatics* 17(1), 62–71.

Smith, B., Ashburner, M., Rosse, C., et al. 2007. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology* 25(11), 1251–5.

Smoot, M.E., Ono, K., Ruscheinski, J., et al. 2011. Cytoscape 2.8: new features for data integration and network visualization. *Bioinformatics (Oxford, England)* 27(3), 431–2.

Snoep, J.L. and Olivier, B.G. 2003. JWS online cellular systems modelling and microbiology. *Microbiology (Reading, England)* 149(Pt 11), 3045–7.

Stanford, N.J., Lubitz, T., Smallbone, K., et al. 2013. Systematic construction of kinetic models from genome-scale metabolic networks. *PloS one* 8(11), e79195.

Stoker, H.S. 2012. *General, Organic, and Biological Chemistry*. Textbooks Available with Cengage YouBook Series. Brooks/Cole.

Sugiyama, K., Tagawa, S., and Toda, M. 1981. Methods for Visual Understanding of Hierarchical System Structures. *IEEE Transactions on Systems, Man, and Cybernetics* 11(2), 109–125.

Swainston, N., Smallbone, K., Mendes, P., et al. 2011. The SuBliMinaL Toolbox: automating steps in the reconstruction of metabolic networks. *Journal of integrative bioinformatics* 8(2), 186.

Tamassia, R. 2007. *Handbook of Graph Drawing and Visualization (Discrete Mathematics and Its Applications)*. Chapman & Hall/CRC.

The UniProt Consortium 2013. Update on activities at the Universal Protein Resource (UniProt) in 2013. *Nucleic acids research* 41(Database issue), D43–7.

Thiele, I. and Palsson, B.O. 2010. A protocol for generating a high-quality genome-scale metabolic reconstruction. *Nature protocols* 5(1), 93–121.

Thiele, I., Swainston, N., Fleming, R.M.T., et al. 2013. A community-driven global reconstruction of human metabolism. *Nature biotechnology* 31(5), 419–25.

Thiele, I., Vlassis, N., and Fleming, R.M.T. 2014. fastGapFill: efficient gap filling in metabolic networks. *Bioinformatics (Oxford, England)* 30(17), 2529–2531.

Thiele, I., Vo, T.D., Price, N.D., et al. 2005. Expanded metabolic reconstruction of Helicobacter pylori (iIT341 GSM/GPR): an in silico genome-scale characterization of single- and double-deletion mutants. *Journal of bacteriology* 187(16), 5818–30.

Tohsato, Y., Matsuda, H., and Hashimoto, A. 2000. A multiple alignment algorithm for metabolic pathway analysis using enzyme hierarchy. *Proc Int Conf Intell Syst Mol Biol.* 8, 376–83.

Umeton, R., Nicosia, G., and Dewey, C.F. 2012. OREMPdb: a semantic dictionary of computational pathway models. *BMC bioinformatics* 13 Suppl 4, S6.

Unwin, A., Theus, M., and Hofmann, H. 2006. *Graphics of Large Datasets: Visualizing a Million*. Springer.

von Landesberger, T., Kuijper, A., Schreck, T., et al. 2011. Visual Analysis of Large Graphs: State-of-the-Art and Future Research Challenges. *Computer Graphics Forum* 30(6), 1719–1749.

Wagner, A. 2012. Metabolic Networks and Their Evolution. *In* O.S. Soyer, N. Back, I.R. Cohen, A. Lajtha, J.D. Lambris, and R. Paoletti, eds., *Evolutionary Systems Biology*, vol. 751 of *Advances in Experimental Medicine and Biology*, 29–52. Springer New York.

Waltemath, D., Adams, R., Bergmann, F., et al. 2011. Reproducible computational biology experiments with SED-ML - The Simulation Experiment Description Markup Language. *BMC Systems Biology* 5(1), 198.

Whetzel, P.L., Noy, N.F., Shah, N.H., et al. 2011. BioPortal: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. *Nucleic acids research* 39(Web Server issue), W541–5.

Yates, T., Okoniewski, M.J., and Miller, C.J. 2008. X:Map: annotation and visualization of genome structure for Affymetrix exon array analysis. *Nucleic acids research* 36(Database issue), D780–6.

Zhukova, A. and Sherman, D.J. 2014a. Knowledge-based generalization of metabolic models. *Journal of computational biology : a journal of computational molecular cell biology* 21(7), 534–47.

Zhukova, A. and Sherman, D.J. 2014b. Knowledge-based generalization of metabolic networks: a practical study. *Journal of Bioinformatics and Computational Biology* 12(2), 1441001.

Zhukova, A. and Sherman, D.J. 2014c. Mimoza. URL http://mimoza.bordeaux.inria.fr/.

Zhukova, A. and Sherman, D.J. 2015. Mimoza: Web-Based Semantic Zooming and Navigation in Metabolic Networks. *BMC Systems Biology* Forthcoming.

Appendix A — Applications of knowledge-based generalization

Table 6.1 Performance of the model generalization method on 269 genome-scale metabolic models.

Model	Number of of reactions (initial model)	Number of of reactions (generalized model)	Compression rate
BMID000000140205	4010	3469	1.16
BMID000000140206	3631	3180	1.14
BMID000000140207	889	801	1.11
BMID000000140208	2366	1989	1.19
BMID000000140209	4088	3576	1.14
BMID000000140210	4597	3985	1.15
BMID000000140211	2868	2538	1.13
BMID000000140212	3887	3367	1.15
BMID000000140213	3824	3329	1.15
BMID000000140214	2678	2447	1.09
BMID000000140215	1280	1147	1.12
BMID000000140216	3319	2924	1.14
BMID000000140217	3651	3204	1.14
BMID000000140218	2209	1890	1.17
BMID000000140219	2768	2279	1.21
BMID000000140220	4162	3665	1.14
BMID000000140221	2027	1921	1.06
BMID000000140222	2550	2155	1.18
BMID000000140223	2970	2632	1.13

Continued on next page

Table 6.1 – continued from previous page

Model	Number of of reactions (initial model)	Number of of reactions (generalized model)	Compression ratio
BMID000000140224	1723	1525	1.13
BMID000000140225	2271	1944	1.17
BMID000000140226	3108	2794	1.11
BMID000000140227	4635	3955	1.17
BMID000000140228	1582	1404	1.13
BMID000000140229	3023	2670	1.13
BMID000000140230	2192	1932	1.13
BMID000000140231	968	891	1.09
BMID000000140232	371	328	1.13
BMID000000140233	3856	3306	1.17
BMID000000140234	2527	2158	1.17
BMID000000140235	1840	1589	1.16
BMID000000140236	3555	3095	1.15
BMID000000140237	1365	1228	1.11
BMID000000140238	3960	3476	1.14
BMID000000140239	2588	2197	1.18
BMID000000140240	659	593	1.11
BMID000000140241	3168	2763	1.15
BMID000000140242	3203	2835	1.13
BMID000000140243	3893	3425	1.14
BMID000000140244	4325	3785	1.14
BMID000000140245	4387	3858	1.14
BMID000000140246	4437	3837	1.16
BMID000000140247	4506	3917	1.15
BMID000000140248	4156	3612	1.15
BMID000000140249	1993	1797	1.11
BMID000000140250	2213	1933	1.14
BMID000000140251	3034	2733	1.11
BMID000000140252	3374	2953	1.14
BMID000000140253	2469	2069	1.19
BMID000000140254	1326	1196	1.11

Continued on next page

Table 6.1 – continued from previous page

Model	Number of of reactions (initial model)	Number of of reactions (generalized model)	Compression ratio
BMID000000140255	3956	3430	1.15
BMID000000140256	2665	2378	1.12
BMID000000140257	3363	2823	1.19
BMID000000140258	2913	2513	1.16
BMID000000140259	2008	1784	1.13
BMID000000140260	4608	3983	1.16
BMID000000140261	1293	1187	1.09
BMID000000140262	3064	2696	1.14
BMID000000140263	3705	3260	1.14
BMID000000140264	2435	2051	1.19
BMID000000140265	2281	2066	1.1
BMID000000140266	2623	2398	1.09
BMID000000140267	1811	1598	1.13
BMID000000140268	4321	3707	1.17
BMID000000140269	3571	2983	1.2
BMID000000140270	2199	1858	1.18
BMID000000140271	3941	3447	1.14
BMID000000140272	2354	2071	1.14
BMID000000140273	2010	1789	1.12
BMID000000140274	1960	1810	1.08
BMID000000140275	3277	2945	1.11
BMID000000140276	3333	2978	1.12
BMID000000140277	4625	4030	1.15
BMID000000140278	987	933	1.06
BMID000000140279	4060	3641	1.12
BMID000000140280	2090	1837	1.14
BMID000000140281	2474	2294	1.08
BMID000000140282	1667	1479	1.13
BMID000000140283	1680	1504	1.12
BMID000000140284	3887	3388	1.15
BMID000000140285	3376	2894	1.17

Continued on next page

Table 6.1 – continued from previous page

Model	Number of of reactions (initial model)	Number of of reactions (generalized model)	Compression ratio
BMID000000140286	2752	2429	1.13
BMID000000140287	4095	3580	1.14
BMID000000140288	3799	3266	1.16
BMID000000140289	4336	3676	1.18
BMID000000140290	2041	1774	1.15
BMID000000140291	4089	3578	1.14
BMID000000140292	3482	2922	1.19
BMID000000140293	3836	3392	1.13
BMID000000140294	3880	3381	1.15
BMID000000140295	1481	1339	1.11
BMID000000140296	3107	2783	1.12
BMID000000140297	3799	3318	1.14
BMID000000140298	2358	2102	1.12
BMID000000140299	1963	1700	1.15
BMID000000140300	2796	2512	1.11
BMID000000140301	1203	1110	1.08
BMID000000140302	406	366	1.11
BMID000000140303	3145	2748	1.14
BMID000000140304	3740	3289	1.14
BMID000000140305	1640	1502	1.09
BMID000000140306	2058	1839	1.12
BMID000000140307	2732	2475	1.1
BMID000000140308	1648	1459	1.13
BMID000000140309	1168	1082	1.08
BMID000000140310	3888	3429	1.13
BMID000000140311	1673	1534	1.09
BMID000000140312	2826	2469	1.14
BMID000000140313	5056	4428	1.14
BMID000000140314	1425	1319	1.08
BMID000000140315	1116	1036	1.08
BMID000000140316	2138	1950	1.1

Continued on next page

Table 6.1 – continued from previous page

Model	Number of of reactions (initial model)	Number of of reactions (generalized model)	Compression ratio
BMID000000140317	3535	2972	1.19
BMID000000140318	1519	1363	1.11
BMID000000140319	2117	1927	1.1
BMID000000140320	2531	2269	1.12
BMID000000140321	3513	3071	1.14
BMID000000140322	4339	3716	1.17
BMID000000140323	597	541	1.1
BMID000000140324	1245	1156	1.08
BMID000000140325	2513	2334	1.08
BMID000000140326	2607	2399	1.09
BMID000000140327	2244	1930	1.16
BMID000000140328	974	872	1.12
BMID000000140329	3231	2880	1.12
BMID000000140330	2011	1828	1.1
BMID000000140331	1693	1542	1.1
BMID000000140332	4269	3669	1.16
BMID000000140333	1633	1509	1.08
BMID000000140334	3546	3107	1.14
BMID000000140335	1650	1516	1.09
BMID000000140336	1928	1771	1.09
BMID000000140337	4316	3703	1.17
BMID000000140338	1548	1377	1.12
BMID000000140339	1879	1697	1.11
BMID000000140340	656	635	1.03
BMID000000140341	2302	1937	1.19
BMID000000140342	3103	2699	1.15
BMID000000140343	2655	2402	1.11
BMID000000140344	1787	1687	1.06
BMID000000140345	3189	2682	1.19
BMID000000140346	1921	1778	1.08
BMID000000140347	2999	2675	1.12

Continued on next page

Table 6.1 – continued from previous page

Model	Number of of reactions (initial model)	Number of of reactions (generalized model)	Compression ratio
BMID000000140348	1930	1818	1.06
BMID000000140349	2895	2627	1.1
BMID000000140350	1799	1545	1.16
BMID000000140351	3620	3170	1.14
BMID000000140352	2586	2351	1.1
BMID000000140353	2927	2631	1.11
BMID000000140354	4744	4174	1.14
BMID000000140355	4673	4017	1.16
BMID000000140356	4670	4043	1.16
BMID000000140357	1673	1485	1.13
BMID000000140358	4382	3794	1.15
BMID000000140359	3200	2852	1.12
BMID000000140360	807	767	1.05
BMID000000140361	3400	2970	1.14
BMID000000140362	5819	4948	1.18
BMID000000140363	1311	1166	1.12
BMID000000140364	3185	2785	1.14
BMID000000140365	3962	3454	1.15
BMID000000140366	4107	3571	1.15
BMID000000140367	3490	3092	1.13
BMID000000140368	1738	1628	1.07
BMID000000140369	2317	2004	1.16
BMID000000140370	4068	3565	1.14
BMID000000140371	4272	3782	1.13
BMID000000140372	2109	1834	1.15
BMID000000140373	1259	1137	1.11
BMID000000140374	2952	2547	1.16
BMID000000140375	944	892	1.06
BMID000000140376	828	788	1.05
BMID000000140377	2595	2370	1.09
BMID000000140378	4528	3864	1.17

Continued on next page

Table 6.1 – continued from previous page

Model	Number of of reactions (initial model)	Number of of reactions (generalized model)	Compression ratio
BMID000000140379	4014	3519	1.14
BMID000000140380	1609	1484	1.08
BMID000000140381	4379	3779	1.16
BMID000000140382	1769	1563	1.13
BMID000000140383	2365	2009	1.18
BMID000000140384	3926	3477	1.13
BMID000000140385	3510	3099	1.13
BMID000000140386	4133	3579	1.15
BMID000000140387	3096	2779	1.11
BMID000000140388	2010	1791	1.12
BMID000000140389	3635	3187	1.14
BMID000000140390	2416	2150	1.12
BMID000000140391	2861	2535	1.13
BMID000000140392	3013	2708	1.11
BMID000000140393	1659	1463	1.13
BMID000000140394	3147	2770	1.14
BMID000000140395	3317	2908	1.14
BMID000000140396	2958	2649	1.12
BMID000000140397	2022	1792	1.13
BMID000000140398	2715	2417	1.12
BMID000000140399	2589	2203	1.18
BMID000000140400	2765	2445	1.13
BMID000000140401	3418	3017	1.13
BMID000000140402	2979	2680	1.11
BMID000000140403	3301	2907	1.14
BMID000000140404	3586	3000	1.2
BMID000000140405	1935	1820	1.06
BMID000000140406	2768	2448	1.13
BMID000000140407	2771	2484	1.12
BMID000000140408	4011	3375	1.19
BMID000000140409	3853	3397	1.13

Continued on next page

Table 6.1 – continued from previous page

Model	Number of of reactions (initial model)	Number of of reactions (generalized model)	Compression ratio
BMID000000140410	2787	2531	1.1
BMID000000140411	3029	2651	1.14
BMID000000140412	4639	3967	1.17
BMID000000140413	1939	1668	1.16
BMID000000140414	2805	2528	1.11
BMID000000140415	1289	1181	1.09
BMID000000140416	1608	1422	1.13
BMID000000140417	3099	2768	1.12
BMID000000140418	2859	2603	1.1
BMID000000140419	2059	1787	1.15
BMID000000140420	3833	3330	1.15
BMID000000140421	3042	2756	1.1
BMID000000140422	2131	1843	1.16
BMID000000140423	4512	3900	1.16
BMID000000140424	1711	1545	1.11
BMID000000140425	3729	3235	1.15
BMID000000140426	1176	1086	1.08
BMID000000140427	2551	2160	1.18
BMID000000140428	2253	1935	1.16
BMID000000140429	2765	2491	1.11
BMID000000140430	3734	3351	1.11
BMID000000140431	1276	1184	1.08
BMID000000140432	3914	3395	1.15
BMID000000140433	2725	2362	1.15
BMID000000140434	4294	3661	1.17
BMID000000140435	4395	3765	1.17
BMID000000140436	2958	2614	1.13
BMID000000140437	2704	2474	1.09
BMID000000140438	3824	3391	1.13
BMID000000140439	2996	2686	1.12
BMID000000140440	2371	2172	1.09

Continued on next page

Table 6.1 – continued from previous page

Model	Number of of reactions (initial model)	Number of of reactions (generalized model)	Compression ratio
BMID000000140441	1848	1743	1.06
BMID000000140442	2732	2484	1.1
BMID000000140443	2627	2373	1.11
BMID000000140444	3027	2639	1.15
BMID000000140445	4260	3706	1.15
BMID000000140446	3733	3311	1.13
BMID000000140447	4005	3519	1.14
BMID000000140448	2114	1892	1.12
BMID000000140449	4333	3708	1.17
BMID000000140450	4198	3731	1.13
BMID000000140451	3114	2729	1.14
BMID000000140452	4337	3737	1.16
BMID000000140453	2492	2196	1.13
BMID000000140454	5072	4335	1.17
BMID000000140455	4051	3559	1.14
BMID000000140456	2778	2513	1.11
BMID000000140457	1753	1521	1.15
BMID000000140458	3846	3369	1.14
BMID000000140459	2545	2290	1.11
BMID000000140460	4547	4057	1.12
BMID000000140461	3337	2961	1.13
BMID000000140462	389	347	1.12
BMID000000140463	4895	4216	1.16
BMID000000140464	1078	1030	1.05
BMID000000140465	3114	2791	1.12
BMID000000140466	3546	2963	1.2
BMID000000140467	4355	3745	1.16
BMID000000140468	4418	3823	1.16
BMID000000140469	3563	3189	1.12
BMID000000140470	4095	3573	1.15
BMID000000140471	3551	3209	1.11

Continued on next page

Table 6.1 – continued from previous page

Model	Number of of reactions (initial model)	Number of of reactions (generalized model)	Compression ratio
BMID000000140472	1743	1555	1.12
BMID000000140473	4040	3446	1.17
Average:	2879	2532	1.14